Politecnico di Milano

School of Industrial and Information Engineering
Master of Science – Electrical Engineering

Primary Frequency Regulation Implemented on Home Appliances

Supervisor
Prof. Roberto Ottoboni

Co-Supervisor
Prof. Enrico Ragaini

Candidate
Mattia Castoldi - 927404

Academic Year 2020 – 2021

Finding a suitable
citation for
an engineering thesis.

—Note on Mattia's desk

Acknowledgements

An era of personal growth comes to a conclusion with this paper.
I would like to thank all the people who are or have been close to me, who support me, help me or simply love me for who I am.

A heartfelt thank you to my supervisor Prof. Ottoboni and co-supervisor Prof. Ragaini for their infinite availability and promptness to my every request.

Abstract

The penetration of non-programmable renewable sources in the electricity production system is leading to various problems in the management of electricity grids. One of these is related to the fact that they are not able to guarantee a primary frequency regulation adequate to the network requirements. The result is a reduction of the available primary reserve. The objective of this thesis project is to increase it by controlling domestic loads. Thanks to the collaboration between the companies Elettrotecnica ROLD S.r.l. and Abb S.p.A., I have started to design in the laboratories of the latter a device that can be associated to a generic non-vital household appliance that is able to control its pause and start status in relation to the value of the network frequency that is being measured.

For this type of control has been adapted an Ekip UP Pro produced by Abb and as programming language has been used the Custom Logic, always developed by Abb. To demonstrate the goodness of the control logic that has been implemented in the device and then also the proper functioning of the overall system, have been identified two model appliances of different nature and were conducted tests on them, which have finally given positive results both as regards the control on the primary regulation, both for the discretion towards the end user who uses the appliance.

In order to increase the ease of management of the electrical network and to improve the quality of its parameters, an integrative software tool was also proposed to help the end user in the management of programmable appliances, advising on their start-up time. To program this tool, I used Python, a high-level language.

The work I done so far does not intend to be a point of arrival, but it is proposed to be only the first step towards the construction of a new product that can make futuristic household appliances active parts in the primary frequency regulation.

KEY WORDS: Primary frequency regulation, Appliances, Ekip UP Pro, Custom Logic, Python.

Sommario

La penetrazione delle fonti rinnovabili non programmabili nel sistema di produzione dell'energia elettrica sta comportando problematiche di varia natura nella gestione delle reti elettriche. Una di queste è legata al fatto che esse non riescono a garantire una regolazione primaria di frequenza adeguata ai requisiti di rete, con conseguente riduzione della riserva primaria disponibile. L'obiettivo di questo progetto di tesi è quello di aumentarla controllando i carichi domestici. Grazie alla collaborazione tra le compagnie Elettrotecnica ROLD S.r.l. e Abb S.p.A., ho iniziato a progettare nei laboratori di quest'ultima un dispositivo associabile a un elettrodomestico non vitale generico che sia capace di controllarne il suo stato di pausa e avvio in relazione al valore che si sta misurando della frequenza di rete.

Per questo tipo di controllo si è adattato un Ekip UP Pro prodotto dalla Abb e come linguaggio di programmazione si è utilizzato il Custom Logic, sviluppato sempre dalla Abb. Per dimostrare la bontà della logica di controllo che si è implementata nel dispositivo e quindi anche il corretto funzionamento del sistema complessivo, si sono individuati due elettrodomestici modello di diversa natura e sono stati condotti dei test su di essi, i quali hanno dato infine esito positivo sia per quanto concerne il controllo sulla regolazione primaria, sia per la discrezione nei confronti dell'utente finale che utilizza l'elettrodomestico.

Per aumentare la facilità della gestione della rete elettrica e per migliorarne la qualità dei suoi parametri, si è inoltre proposto uno strumento software integrativo che aiuti l'utente finale nella gestione degli elettrodomestici programmabili, consigliandone il loro tempo di avvio. Per la sua programmazione ho utilizzato Python, un linguaggio ad alto livello.

Il lavoro che ho fin qui condotto non intende per nulla essere un punto di arrivo, ma si propone invece di essere solo il primo passo verso la costruzione di un nuovo prodotto che possa rendere avveniristicamente gli elettrodomestici delle parti attive nella regolazione primaria di frequenza.

PAROLE CHIAVE: Regolazione primaria di frequenza, Elettrodomestici, Ekip UP Pro, Custom Logic, Python.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Background

In recent years, the issue of environmental sustainability of the energy sector is becoming increasingly relevant, as the growing influence of human activity on the Earth's temperature is largely due to the use of fossil fuels to generate electricity and for mobility. This effect is produced by the enormous quantities of greenhouse gases released into the atmosphere, in addition to those naturally present, giving rise to the so-called "greenhouse effect" and therefore to global warming. All major government agencies are trying to find solutions to this problem. The EU itself has set targets up to 2050, in which it envisions a totally zero-emissions future, to progressively reduce greenhouse gas emissions. The main objective is to provide a guide for including pollution prevention in all relevant EU policies, maximising synergies in an effective and proportionate way, stepping up implementation and identifying possible gaps or trade-offs. This action plan sets key targets on 2030 to speed up pollution reduction. By 2030 in fact the EU should reduce:

- By more than 55% the health impacts (premature deaths) of air pollution;
- By 30% the share of people chronically disturbed by transport noise;
- By 25% the EU ecosystems where air pollution threatens biodiversity;
- By 50% nutrient losses, the use and risk of chemical pesticides, the use of the more hazardous ones, and the sale of antimicrobials for farmed animals and in aquaculture;
- By 50% plastic litter at sea and by 30% microplastics released into the environment;
- Significantly total waste generation and by 50% residual municipal waste. [1]

These goals can only be achieved by replacing the way in which most energy has traditionally been produced. Heading in this direction, last year, in 2020, the

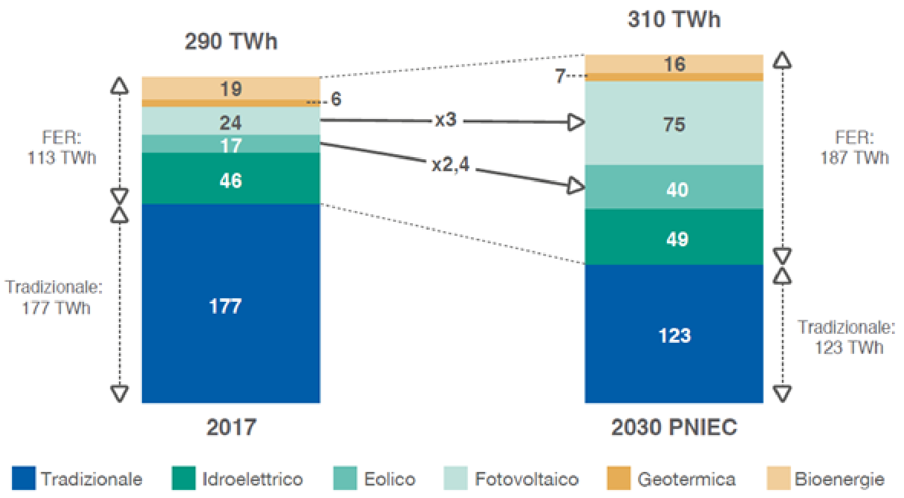

Figure 1.1. Evolution of electricity production to 2030 according to Terna

estimates of growth of sustainable energy obtainable from renewable sources made by analysts at Bloomberg New Energy Finance (BNEF), showed that the global health crisis due to Covid-19 has not stopped the installation of new green energy plant installations. The estimates are in fact particularly good for photovoltaic plants: by the end of this year, new solar installations could exceed 150 GW for the first time, after the 2020 that with 132 GW was still able to beat previous records. Expected also the umpteenth decline in the price of solar modules, which will fall from an average of 20 cents per watt to about 18 cents, thus increasing further expansion for the future. Wind power is also seeing an excellent 2021, with a total of 84 GW of new capacity installed, the majority of which is onshore. Power installed in Europe will amount to around 20 GW, 5 GW more than the previous record, and this will contribute on a global scale, together with new installations in the Americas, to largely offset the expected decline in China. [2]

The installation of this type of plant based on renewable energy, therefore is gradually replacing the old traditional systems of electricity generation as shown in figure 1.1, allowing to avoid greenhouse gas emissions into the atmosphere by exploiting precisely renewable resources such as wind and sun. As previously mentioned, this evolution is essential to achieve the objectives of a zero emission future desired by the EU, but there is a big downside. The negative side is that with their uncontrolled growth, exasperated by very convenient incentives, without an adequate evolution of the electrical system, the management of the electrical distribution network is increasingly difficult.

Another recent phenomenon is the spread of the DG (Distributed Generation), i.e. generation plants connected to the distribution grid, which historically was designed for unidirectional power flows, from the transmission grid to end users. For a few years on, however, there has been a reversal of power flows - from the distribution network to the transmission network - in most substations for more than 5% of the time. The penetration of the DG caused the reduction in the number of thermal

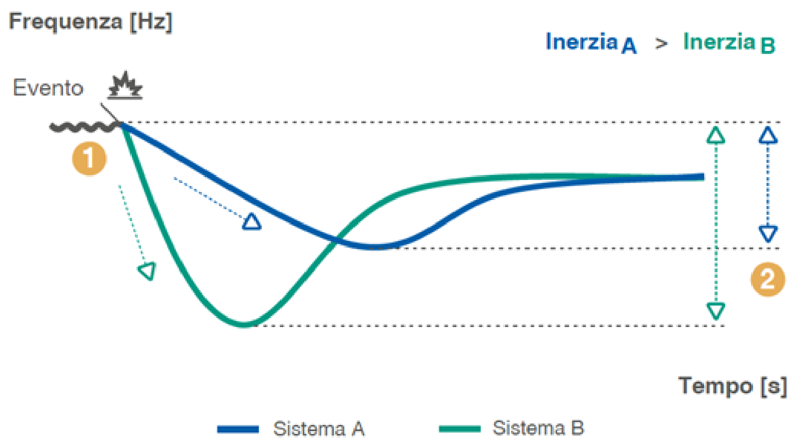

Figure 1.2. Frequency transient trend in case of generation loss on two systems characterized by different inertia

units qualified to offer regulation services. This is causing a gradual decrease in the regulating capacity of the system's primary reserve, with negative impacts on safety and control. Difficulties in the management of the electricity grid in fact have arisen because both renewable plants and the DG are not required to provide grid services such as voltage and frequency regulation. Therefore, there has been a deterioration in the quality of service of the power supplied to end users. Service quality is a fundamental aspect of increasing importance for two main reasons. The first one is the increasing electrification of end-user consumption, which makes continuous service availability essential. The second one is the increasing presence of electronic components for the automation of user systems, especially the industrial ones, which require for their proper functioning a high quality level of electrical energy supply. [3] The situation is even aggravated by the fact that the number of generators connected to the grid through electronic converters without inertia is increasing at the same rate. The inertia, linked to the rotating masses of the turbine/alternator groups, is of fundamental importance for the stability of the system in the first moments after a disturbance as shown in figure 1.2.

As a result, the grid frequency profile will be worse, with more significant and frequent deviations from the 50 Hz nominal value and from the nominal voltage.

Future Purpose

In order to ensure the safe operation of the electrical system, it would be appropriate for the future to provide for a greater participation of plants from renewable sources in grid services. However, one possible solution to mitigate the previous problem could be to consider the use of a variety of non-viable household loads that normally do not provide support at the primary control level, relieving or burdening the power demand from the grid, depending on the state of the measured electrical parameters. Already several theoretical studies and simulations have shown the potential and benefits that can bring the control of a large number of domestic loads to the network operators in terms of power quality and emergency management. [4] [5]

In this thesis work we start from the idea that these studies have suggested, to implement in reality a control of this type, with an independent logic of control at the level of the home network, trying to make it as effective as possible for network operators, but that is also discreet enough not to burden the habits and needs of domestic users. The strong points of this project must be those that, while this technology brings benefits to the consumer himself, it does not alter his habits. Only in this way, in fact, we can hope that there will be a penetration, at least on a national scale, of a device with a technology of this type.

Chapter 2

Primary frequency regulation

2.1 Generalities

The electrical system is divided into the three basic phases of electrical energy:

- Production
- Transmission
- Distribution

Electricity production consists in converting the primary energy source - such as coal, oil, gas, nuclear for traditional sources and wind, solar, hydroelectric for renewable sources - through a production plant. Generally, in traditional large power plants, electrical power is generated at medium voltage (10-20 kV) and through a transformer the voltage level is increased (220-380 kV) and then electricity can be transmitted thanks to the transmission lines. Transmission networks must therefore allow the flow of large powers over long distances, from power plants to consumption areas. [6] On the contrary, renewable energy plants are generally characterized by more contained powers; in this case, power is introduced in the network at medium voltage (20 kV) or low voltage (400 V), with criticalities that will be exposed later on. The distribution networks, such as medium and low voltage mentioned above, and high voltage, have the task of connecting every single user to the electric system. The fundamental constrain of the electrical energy generated by the production plants and transmitted/distributed by the transmission/distribution networks is that it must be consumed instant by instant by the end users - typically industrial or residential loads -. The requirement of electric power, both the active [kW] and the reactive part [kVAR], is therefore associated with an electric load, which is variable instant by instant, but with predictable and similar trends at the same time interval and period considered. Thus, the grid operator predicts the required power demand day by day in order to operate the electrical system in the best possible way to ensure the energy supply to the end users. [7] An example of this forecasting is given in the picture 2.1.

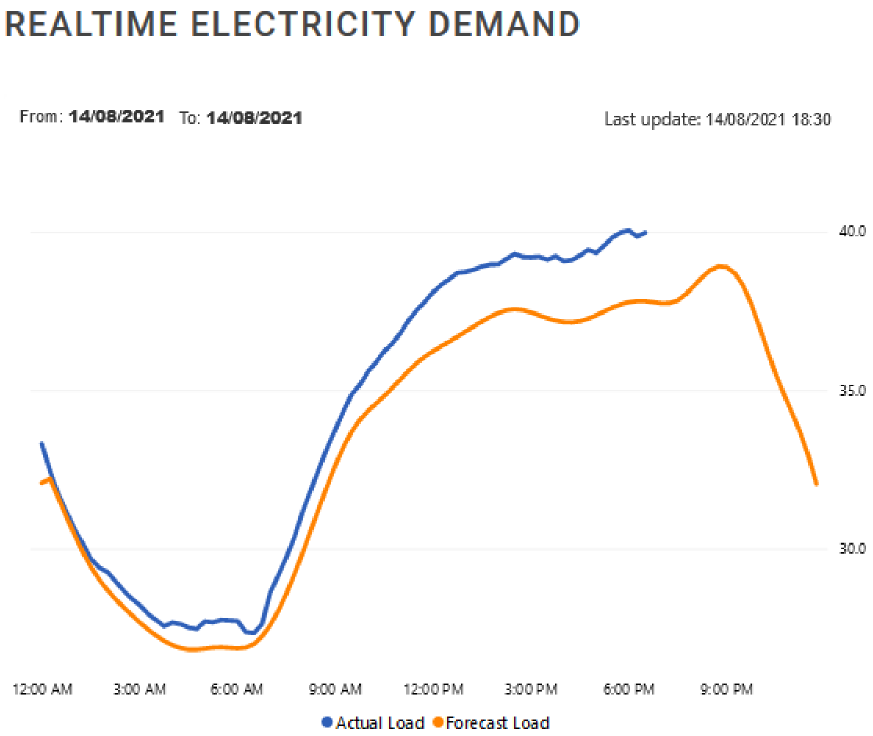

Figure 2.1. Example of forecast and actual load trend

2.2 Notes on voltage regulation

The value assumed by the voltage in an electrical network is an index of the quality of the electrical energy that is supplied to the loads. In fact, all the loads should operate at a voltage close to the one designed for them to ensure their safety of service and their correct operation. The active and reactive powers required by both industrial and residential loads, should flow through the transmission and distribution network without excessive voltage drops. As demonstrated below, the amplitudes of the voltages in the various nodes of an electrical network depend mainly on the flows of reactive power. Therefore, an effective voltage regulation is linked to the presence in the network of devices capable to manage reactive power to compensate the one absorbed by loads and other elements of the network itself.

To understand how voltage regulation occurs, we consider a purely inductive connection as modelled in Figure 2.2, which can represent the reactance of a single network element - e.g. a generator, a transformer, or a line - since generally the resistive component is negligible compared to the inductive one.

The complex power on arrival is equal to:

$$S_a = P_a + Q_a = \frac{v_p v_a}{x} e^{j(\frac{\pi}{2}-\theta)} + \frac{v_a^2}{x} e^{j\frac{\pi}{2}} \tag{2.1}$$

Where the quantities are expressed in per unit (pu) and θ is the angle of displacement

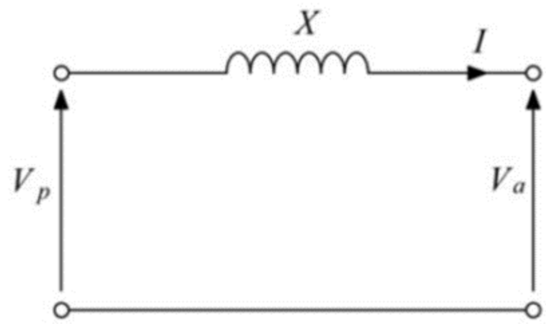

Figure 2.2. Purely inductive connection

between v_p and v_a. Separating the real and imaginary part:

$$P_a = \frac{v_p v_a}{x} sen\theta \tag{2.2}$$

$$Q_a = \frac{v_p cos\theta - v_a}{x} v_a \tag{2.3}$$

It can be seen that, in the case of a purely inductive connection, the active power is related to the phase shift θ, and the reactive power related instead to the amplitude difference between the voltages. Considering θ sufficiently small, it can be assumed that $cos(\theta) \approx 1$ and $sen(\theta) \approx \theta$:

$$P_a \approx \frac{v_p v_a}{x} \theta \tag{2.4}$$

$$Q_a = \frac{\Delta v}{x} v_a \tag{2.5}$$

To conclude, it can be seen that the voltage drop in a transmission line is mainly due to circulating reactive power. Therefore, it is recommended to feed reactive power into the grid on nodes close to where it is required. [6]

2.3 Notes on frequency regulation

Electrical power systems are generally interconnected systems, as shown in Figure 2.3, where all the generators work in parallel and with the help of the distribution networks all the end-users of all the systems are supplied with power at the same frequency. [8]

The frequency value is a parameter of fundamental importance, and it is strictly related to the active powers transmitted from the generators to the loads. At each instant, in fact, the generated power must equal the power absorbed by the loads - plus the inevitable losses -; otherwise, there will be a deviation from the nominal frequency value, which is equal e.g. to 50 Hz for the entire European electrical system. This

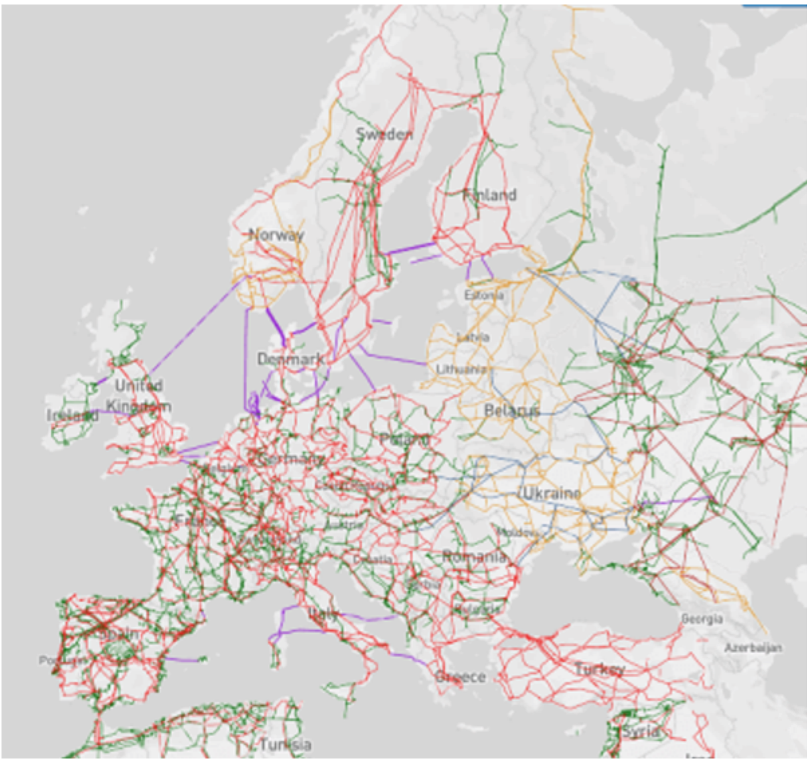

Figure 2.3. Main connections in the European electricity grid

deviation just mentioned above will be greater, if the difference between generation and absorption is greater; for example, it will be low for load fluctuations, large for power plant losses. In case of excess generation, or decrease in load, there is an increase in frequency with respect to the nominal value, and action will be taken by reducing the generated power. Conversely, an increase in the power absorbed by the utilities causes a decrease in the speed of synchronous generators connected in parallel to the network, and therefore the frequency tends to drop, according to the relationship:

$$f = \frac{pn}{60} \tag{2.6}$$

Where p is the number of polar pairs and n is the number of revolutions per minute of the machine. Similarly, a sudden loss of generation will cause a decrease in frequency, and the power plants still in service will have to cope with the missing power share to meet the demands of the loads. Frequency variations must be very limited, firstly to allow proper operation of the loads; think, for example, of an asynchronous motor that at reduced frequency works at lower efficiency and at lower speed than desired. Secondly, the generating stations themselves cannot operate continuously at a frequency less than 48 or 48.5 Hz because of the auxiliary services driven by synchronous generators, e.g. boiler water supply, fuel, combustion, etc. Normally, the grid operator is able to contain frequency deviations to within $\pm$ 0.05 Hz, except for severe disturbances. [6]

The theory of frequency regulation is presented in the following section, since it is the only one of interest for this thesis work - specifically primary frequency regulation -.

2.4 Frequency regulation theory

As can be seen from the premise of the previous section, following an imbalance between the active power generated and the one absorbed by the loads, a transient frequency variation begins, depending on the entity of the imbalance itself, and it is necessary to intervene by acting on the active power supplied by the generators themselves. The frequency regulation can be divided into three phases:

- Primary regulation
- Secondary regulation
- Tertiary regulation

To explain the frequency regulation operations and highlight the quantities of interest, we consider a step increase ΔP in the active power demanded by the loads.

2.5 Primary regulation

The primary frequency regulation starts when there is a disturbance that involves a difference between the generated power and the absorbed power, with consequent variation of the frequency. Following this disturbance, there is a variation in the power absorbed by the loads and the intervention of the speed regulators and therefore of the active power fed into the network by the production units to adjust the power balance. At the end of the primary regulation there is a new steady state condition, in which generated and absorbed power are in balance, but the value of the network frequency is different from the one before the disturbance. The role of primary regulation, in fact, is not to bring the frequency back to the nominal value, but it is to try to contain as much as possible its deviation and to maintain the balance between generation and power demand in the electrical system.

To analyze how frequency varies over this time interval, and to study its active power balance we consider a single generating unit, consisting on a turbine-alternator system - for simplicity of unit efficiency -, and a system of loads. [6] In the absence of disturbances, the power P_i fed into the turbine is equivalent to the power P_u absorbed by the loads. At the time of the step disturbance ΔP, a relevant factor in the frequency transients is the change in kinetic energy of rotating machines connected to the grid - i.e. synchronous and asynchronous generators and motors - due to the change in their rotational speed. The kinetic energy of electrical machines opposes the frequency variation since they tend to maintain their speed. It means that the derivative of kinetic energy corresponds to an accelerating power absorbed or given up by rotating machines depending on whether their speed is increasing or decreasing.

$$\Delta P_a(t) = \frac{dW_{cin}}{dt} \tag{2.7}$$

Thus following the disturbance ΔP, a perturbed regime will begin and the power balance results:

$$P_i^* + \Delta P_i(t) - \Delta P_{a1}(t) = P_u^* + \Delta P_u(t) + \Delta P_{aU}(t) + \Delta P(t) \tag{2.8}$$

Where:

- ΔP_i is the change in power of the generating plant as a result of the intervention of the speed controller, which is caused by the change in frequency;
- ΔP_{a1} is the accelerating power change of the generating plant;
- ΔP_u is the change in power absorbed by the loads as a result of the frequency change;
- ΔP_{aU} is the accelerating power variation of the loads connected to the network.

The negative sign of ΔP_{a1} is due to the fact that, for example, if the frequency drops, the change in kinetic energy is negative; i.e. the accelerating power provides a positive power contribution, helping to contain the frequency deviation. Similar discussion applies to the ΔP_{aU} contribution. Given that $P_i^* = P_u^*$ and merging the two contributions of kinetic energy variation $\Delta P_{a1} + \Delta P_{aU} = \Delta P_a = \frac{dW_{cin}}{dt}$, we obtain:

$$\Delta P_i(t) - \Delta P(t) = \Delta P_u(t) + \Delta P_a(t) \tag{2.9}$$

To define the network transfer function in primary regulation, we must first define $\Delta P_u(t)$ and $\Delta P_a(t)$.

To obtain $\Delta P_u(t)$ we assume that the power absorbed by the loads as a function of frequency is:

$$P_U(t) = P_U^* \left(\frac{f(t)}{f^*}\right)^{\alpha} \tag{2.10}$$

Valid for small changes in frequency f from the initial value f^*. The exponent α is typically between 1,5 and 2. This derives from the fact that many utilities are made up of asynchronous or synchronous motors, which must supply mechanical loads with a greater torque as the speed increases, which is imposed by the network frequency value. For example, for centrifugal pumps $\alpha = 3,4$; for fans $\alpha = 2,3$; others $\alpha = 1$ or null. Deriving with respect to f:

$$\frac{dP_U}{df} = P_U^* \alpha \left(\frac{f}{f^*}\right)^{\alpha-1} \frac{1}{f^*} \tag{2.11}$$

The frequency variation generally stays within modest limits, so $\frac{f}{f^*} \approx 1$ can be considered. Then for finite increments of the frequency we get:

$$\Delta P_U(t) = \frac{\alpha P_U^*}{f^*}\Delta f(t) = K_U \Delta f(t) \tag{2.12}$$

The K_U parameter represents the regulating energy of the loads in Joule.

Regarding the kinetic energy of rotating machines, and hence the contribution ΔP_a, we observe that the kinetic energy stored by all rotating machines depends on the square of the angular velocity ω, and is equal to:

$$W_{cin} = W^* \left(\frac{f(t)}{f^*}\right)^2 \tag{2.13}$$

Derives with respect to time:

$$\frac{d}{dt}W_{cin} = 2W^* \frac{f(t)}{f^*}\frac{1}{f^*}\frac{df}{dt} \tag{2.14}$$

Considering that:

$$\frac{df}{dt} = \frac{d}{dt}(f^* + \Delta f) = \frac{d}{dt}\Delta f \tag{2.15}$$

and that $\frac{f}{f^*} \approx 1$, it results:

$$\frac{d}{dt}W_{cin} = \frac{2W^*}{f^*}\frac{d}{dt}\Delta f = \Delta P_a(t) \tag{2.16}$$

Finally, by defining $K_W = \frac{2W^*}{f^*}$:

$$\Delta P_a(t) = K_W \frac{d}{dt}\Delta f \tag{2.17}$$

The parameter K_W can be defined as a function of the starting time of a unit of rated power P_N, rated speed Ω_N and total moment of inertia J, which is supposed to be started by means of a constant torque equal to the rated C_N, from standstill, up to the rated speed.

$$C_N = \frac{P_N}{\Omega_N} = J\frac{d\Omega}{dt} \tag{2.18}$$

$$T_a = \int_0^{T_a} dt = \frac{J\Omega_N}{P_N}\int_0^{\Omega_N} d\Omega = \frac{2W_N}{P_N} \tag{2.19}$$

with

$$W_N = \frac{1}{2}J\Omega_N^2 \tag{2.20}$$

That said, $2W^*$ can be indicated by distinguishing between the start-up time T_{aN} of the generator and T_{ar} of a generic engine:

$$2W^* = P_N T_{aN} + \sum_r P_r T_{ar} = P_N \left(T_{aN} + \frac{1}{P_N}\sum_r P_r T_{ar}\right) = P_N T_a \tag{2.21}$$

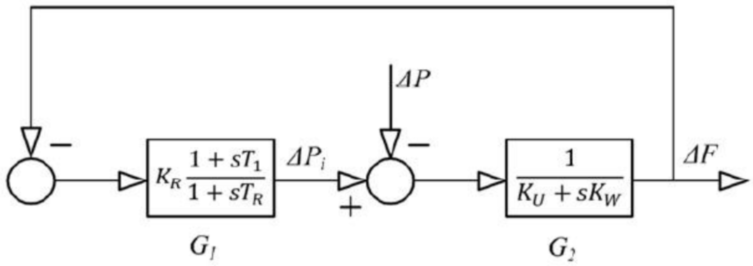

Figure 2.4. Block diagram of primary regulation

With T_a defined as network startup time. So we can specify K_W as:

$$K_W = \frac{P_N T_a}{f^*} \tag{2.22}$$

At this point we can substitute $\Delta P_U(t)$ and $\Delta P_a(t)$ on the equation (2.9), getting:

$$\Delta P_i(t) - \Delta P(t) = K_U \Delta f(t) + K_W \frac{d}{dt}\Delta f \tag{2.23}$$

At this point, we switch to the frequency domain through the use of the Laplace transform, obtaining:

$$\Delta P_i - \Delta P = \Delta F(K_U + sK_W) \tag{2.24}$$

Thus, the change in frequency is given by:

$$\Delta F = \frac{1}{(K_U + sK_W}(\Delta P_i - \Delta P) \tag{2.25}$$

The only value not yet defined is the contribution ΔP_i, which depends on the speed controller and on the characteristics of the production plant. In the primary regulation, the position of the speed controller remains unchanged, and there is only the accelerometric contribution resulting from the frequency offset, which acts on the turbine distributor, varying the power fed into the grid by the value ΔP_i. The transfer function that links ΔP_i to ΔF results:

$$\Delta P_i = K_R \frac{1 + sT_1}{1 + sT_R}(-\Delta F) \tag{2.26}$$

The block diagram of the primary control is shown in Figure ??.

Denoting by G1 and G2 the transfer functions on the diagram, we obtain the block diagram represented in Figure ??.

Where G_N is the transfer function of the network in primary regulation, in which the parameters concerning loads, rotating machines, generator group and relative regulator - with blocked variable speed drive - intervene. To evaluate G_N it is necessary to make explicit G_1 and G_2. We obtain:

$$G_N = \frac{1 + sT_R}{(1 + sT_R)(K_U + sK_W) + K_R(1 + sT_1)} = \frac{1}{T_R K_W} \frac{1 + sT_R}{s^2 + s\frac{K_U T_R + K_W + K_R T_1}{T_R K_W} + \frac{K_R + K_U}{T_R K_W}} \tag{2.27}$$

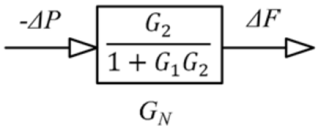

Figure 2.5. Simplified block diagram of primary regulation

We verify the stability of the system by evaluating the roots of the characteristic equation. It is defined:

$$\omega_0 = \sqrt{\frac{K_R + K_U}{T_R K_W}} \tag{2.28}$$

$$\xi = \frac{1}{2}\frac{K_U T_R + K_W + K_R T_1}{\sqrt{T_R K_W (K_R + K_U)}} \tag{2.29}$$

So it can be written:

$$G_N = \frac{1}{K_R + K_U}\frac{\omega_0^2 + \omega_0^2 T_R s}{s^2 + 2\xi\omega_0 s + \omega_0^2} \tag{2.30}$$

The characteristic equation and its roots are:

$$s^2 + 2\xi\omega_0 s + \omega_0^2 = 0 \tag{2.31}$$

$$\alpha_{1,2} = -\omega_0\xi \pm \omega_0\sqrt{\xi^2 - 1} \tag{2.32}$$

The roots have a negative real part, except for $\xi = 0$, showing stability of the setting and highlighting that the variation of the network frequency may have aperiodic or damped oscillatory pattern, depending on whether ξ is greater or less than 1. The response $\Delta f(t)$ following a disturbance $\Delta P(t)$ is obtained by anti-transforming:

$$\Delta F = \frac{1}{K_R + K_U}\frac{\omega_0^2 + \omega_0^2 T_R s}{s^2 + 2\xi\omega_0 s + \omega_0^2}(-\Delta P) \tag{2.33}$$

Applying the final value theorem, it is inferred that following a step disturbance ΔP, once the transient is over, the system will exhibit a frequency change equal to:

$$\Delta f = \frac{-\Delta P}{K_R + K_U} \tag{2.34}$$

This relationship shows that the action of the regulator allows to contain much more effectively the frequency deviation, being normally $K_R \gg K_U$. The parameter $K_N = K_R + K_U$ is defined as the regulating energy of the network.

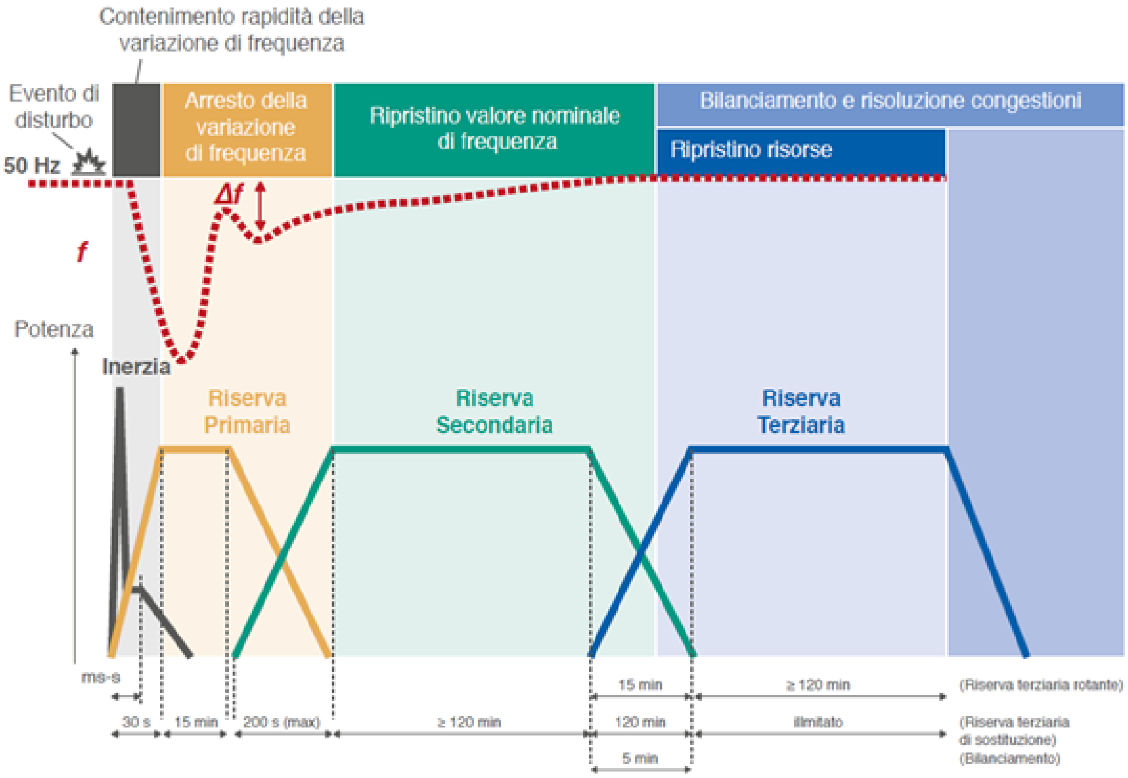

Figure 2.6. Schematic diagram of frequency regulation as a function of the switching time

2.6 Notes on secondary and tertiary regulation

It has just been seen that the role of primary regulation is to re-establish the balance between required and generated power, to limit in this way the frequency variation of the electrical system. Bringing the frequency back to its nominal value is the task of secondary regulation, which intervenes with longer times, once primary regulation has been completed, and it is carried out by varying the power of the groups according to the difference Δf obtained. In the past, in order to vary the power supplied by the generator, it was necessary to act manually on the variable speed drive, while nowadays the regulation is automatic, through the acquisition and the processing of a signal called “regulation level”, sent by an automatic centralized device, the network regulator. Tertiary regulation, unlike the previous ones, is carried out at the request of the network operator, for example with the entry into service of a reserve production plant, or with the variation of the power of one already in service. [9] Secondary and tertiary regulation will not be explored further, since only primary frequency regulation is taken into account for the project developed in this thesis. However, in the figure 2.6 are summarized the times of intervention of the three frequency regulations.

Chapter 3

Project description

3.1 General purpose and personal contribution

This thesis is nothing more than the result of my work in the company Abb S.p.A. and the synergy with which it and Elettrotecnica Rold S.r.l have started to collaborate together. Their initial intentions were to combine the skills that each of them possesses to be able to design a new device suitable for monitoring and controlling the consumption of household loads in relation to the parameters of the electrical network. To achieve this goal, the company Abb Sace has undertaken to provide me with its software and hardware systems for monitoring and management of electrical systems; the company Rold has instead provided me with logistical support, consultation and manpower in the identification, assembly and - taking into account the electrical control systems used by the instrumentation of Abb Sace - modification of prototypes that I used as modeling of domestic loads for laboratory tests. For this project the two companies have given me wide margins of autonomy and freedom in managing both the resources made available to me by them and the instruments present in their laboratories. Thus, in this paper, I trace the steps I have personally taken towards the construction of a new device for primary frequency regulation, operating not on the generation system, but directly on the users one.

3.2 Device designed

The device that has been assembled, and that will be treated more specifically in the following chapters, is an electronic component that can be programmed to be associated with a single load that in relation to the network frequency that it directly measures, can act on the set temperature or on the start and stop command of the appliance, controlling therefore its power consumption. At logical level it is based on the intersection of times that start from the occurrence of certain events, as for instance the exceeding of a threshold on the frequency value. However, due to the fact that it is an independent control device without any external communication, it is not possible in the future to put on the market this type of devices with the same thresholds and the same time curves of intervention. In fact, supposing that commercially this device is quite successful, being able to obtain a more or less wide penetration in the electric system, at the occurrence of a specific event on the electric network, this would simultaneously trip the intervention of all the control devices, causing the switching on or off of a large number of loads. This event would cause a request for electrical power with a variation of slope too steep with respect to a time span too short and therefore difficult to manage by the generation plants that could already be in difficulty for the management of another emergency. In this study that has been carried out, for the control of the prototypes, I have developed two main logic structures and I gave some intervention times that can be considered as "minimum", to guarantee a greater rapidity of intervention on the frequency regulation. Later, however, these preset times, from device to device, will be varied randomly but respecting a preset distribution curve to ensure that the set of loads that have integrated this technology, has overall a correct operation on frequency regulation. The study of this distribution and the related theoretical simulations that will result are however left to future research.

Chapter 4

Control logics already theorized

4.1 Theoretical hypothesis

The idea of this thesis project is based on theoretical studies discussed in some papers available in literature. Historically, it has always been thought that in the behavior of a distribution system it should always be the generators to adapt the power produced to the power demanded by loads and therefore only generators can have an active role in controlling the network parameters. Recently instead, due to the evolution of the general characteristics of the electrical network and to the new possibilities that technological development can offer today, a new scenario has opened up in which even the inert components of the network could have an active role in the control of the network itself. [10] Generally, the control that we are trying to implement in the near future is a frequency control, that consists in forcing the switching on or off of a predetermined number of appliances in relation to the deviation from the nominal value of the network frequency. Of course, there are a lot of possible ways to implement a control logic like this. For example, the European Network of Transmission System Operators for Electricity (ENTSO-E) proposed a strategy of load management for primary frequency regulation supported by thermostatically controlled loads, based on an algorithm that takes their internal temperature into account. In practice, in according to this control logic, if the main frequency is lowered, the preset temperature band of a thermal household appliance, such as a refrigerator, moves proportionally upwards, to higher temperatures, thus forcing the switching off of a number of refrigerators statistically proportional to the lowering of the frequency. If the frequency rises instead the band moves downwards and forces the switching on. However, the band shift does not occur for minimal frequency differences, but only if a certain deviation from the nominal frequency is exceeded (e.g. $\pm$ 0.1 Hz). [11]

This type of control proposed, however, can be optimized with slightly more complex algorithms, which have already been proposed and tested by simulating them in realistic scenarios. These more advanced logics that have yielded better results are:

- Strategy 1 No reset.

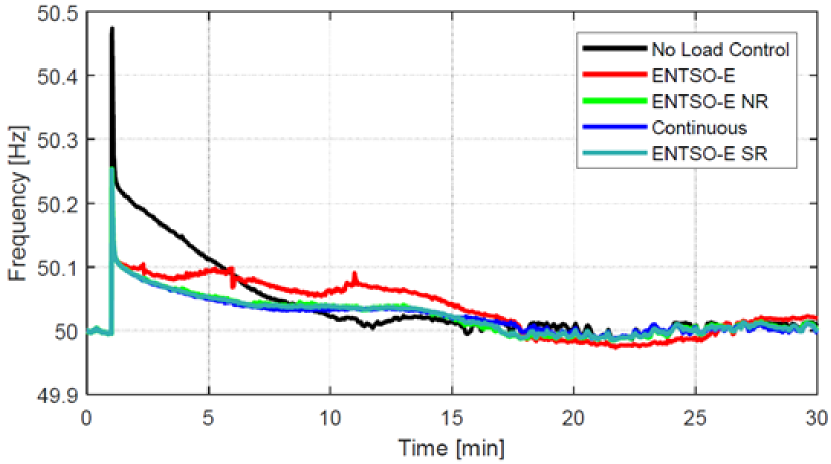

Figure 4.1. Frequency response after the loss of the SACOI line in the Sardinian electrical system

When the frequency returns in the band of values before the offset that activated the frequency control, the control itself does not turn off but continues to remain active (perhaps until the frequency deviation passes through zero).

- Strategy 2 Continuous Control.

  The minimum deviation between the nominal frequency and the threshold that activate the control is considerably decreased (i.e. ± 0.02 Hz).

- Strategy 3 Smooth Recovery.

  When the frequency returns in the band of values before the offset that activated the frequency control, neither turn off nor keep the control active, but vary it more smoothly according to the equation (4.1), where α is the term indicating the proportion between frequency offset and temperature band shift:

$$\alpha' = \alpha \frac{|f - f_{nom}| - 0.02}{\Delta f_{DB} - 0.02} \tag{4.1}$$

  The control is then deactivated when the frequency reaches a very small deviation from the nominal value (i.e. ± 0.02 Hz). [5]

4.2 Simulations

A simulation study, of the Sardinian and Corsican electric system, with the Montecarlo method was conducted. It simulated both a scenario of loss of the SACOI line and a scenario of loss of an entire energy production system - specifically the thermosolar generation -. The benefits that would be brought by being able to control the switching on or off of 1000 sample refrigerators and 1000 sample boilers according to the logics described above are shown in Figure 4.1. [5]

In particular, this graph shows, for the loss scenario of the SACOI line, how the variations of the network frequency would be contained, if using control logics, compared to the absence of control, decreasing the maximum frequency deviation from its nominal value. For the real application of these logics in practice a change of the internal logic of the thermal loads is needed. It would be necessary to act directly on their electronic board, which will probably internally use some communication protocols. Therefore, the fundamental problem, in order to apply one of these proposed and already theoretical tested logics, is to know the internal registers, modify their values and even add one or more register entries for the new data provided by other sensors to be installed, such as the one designed to measure the network frequency.

In order to analyze this problem in more depth and to be able to address it, it is therefore necessary to deepen our knowledge of the internal communication protocols of home appliances. Therefore, in the next chapter we will look at the I2C communication protocol, one of the most widely used communication protocols in this area, to see more clearly how information is exchanged between the different components that make up the electronic infrastructure of any home appliance.

Chapter 5

Internal communication of home appliances

5.1 Notes on I2C

Nowadays all the electronic components inside a generic device supplied with electricity share information with each other by an internal communication protocol. I2C is one of the most widely used protocols of this type within home appliances on the market today. Given its characteristics that we will see in this chapter, indeed, I2C is used to communicate with devices where simplicity and low cost are prioritized over transmission speed. Common applications in fact are for example access to Flash Memory and EEPROM that keep the data stored even when they are not powered or for reading and diagnostics of sensors such as the temperature ones, that are typically inside thermal loads such as refrigerators, boilers or air conditioners. [12] [13] With I2C, you can connect multiple slaves to a single master, or you can have multiple masters controlling single, or multiple slaves. This is really useful when you want to have more than one micro-controller logging data to a single memory card or displaying text to a single LCD. [14] It is a two-wire serial communication system - Figure 5.1 - used between integrated circuits, so I2C only uses two wires to transmit data between devices. These two connections are the SDA (Serial Data), i.e. the line for the master and slave to send and receive data, exactly because I2C is

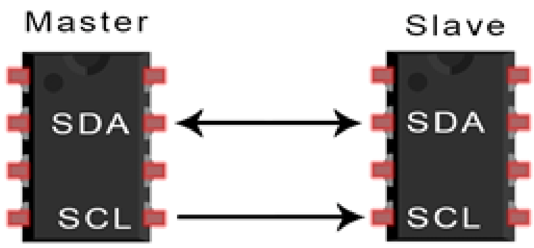

Figure 5.1. Two-wire serial communication: SDS and SCL

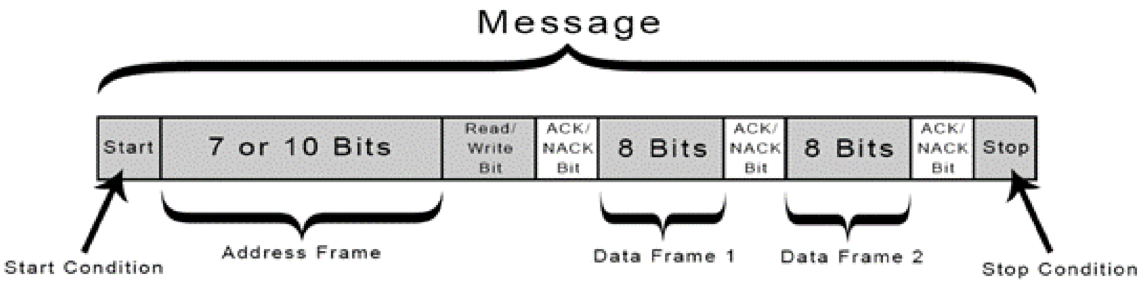

Figure 5.2. Message sent in according to the I2C communication protocol

a serial communication protocol, thus data is transferred bit by bit along this path; and the SCL (Serial Clock), i.e. the line that carries the clock signal. I2C in fact is also a synchronous protocol. This means that the output of bits is synchronized to the sampling of bits by the clock signal, always controlled by the master, and shared between itself and its slaves. In the specific case of our project, to achieve our goal, in a pre-installed communication network, if we want to insert a new element that would interact with the registers delivered between the master and slaves that are still there, we must be sure that its communications are synchronous with the clock of the master. This operation however is not so trivial to achieve in practice and this difficulty must be taken into account for the future.

Data is transferred in messages that are broken up into frames of data. I2C has no slave selection lines, so the SDA line coming out from the master reaches all the SDA pins of the slaves, and every message sent by the master is read by all the slaves. Each message so has an address frame that contains the binary address of the slave the master wants to communicate with. The full message delivered by the master contains the elements shown in Figure 5.2. These elements are:

- The start condition.

  It is communicated when the master wants to interact with someone. The SDA line switches from a high voltage level to a low voltage level before the SCL line switches from high to low.

- The ACK/NACK bit.

  Each frame in a message is followed by an acknowledge/no-acknowledge bit. If an address frame or data frame was successfully received, an ACK bit is returned to the sender from the receiving device.

- The address frame.

  A bit sequence unique to each slave that identifies the slave when the master wants to communicate with it. Usually, a 7 bit address is used and that allow 128 (2^7) unique address. Using instead a 10 bit address - even if in practice it is uncommon - it provides 1,024 (2^{10}) unique addresses and related slaves. [12] After this bit communication, each slave compares the address sent from the master to its own address. If the address matches, it sends a low voltage ACK bit back to the master. If the address doesn't match, the slave does nothing, and the SDA line remains high.

- The read/write bit.

  The address frame includes a single bit at the end that informs the slave whether the master wants to write data or receive data from it. If the master wants to send data to the slave, the read/write bit is a low voltage level. If the master is requesting data from the slave, the bit is a high voltage level.

- The data frame.

  After the master detects the ACK bit from the slave, the first data frame is ready to be sent. The data frame is always 8 bits long, and it is sent with the most significant bit first. Each data frame is immediately followed by an ACK/NACK bit to verify that the frame has been received successfully. The ACK bit must be received by either the master or the slave - depending on who is sending the data - before the next data frame can be sent.

- The stop condition.

  It is used obviously at the end of the message. After all the data frames have been sent, the master can send a stop condition to the slave to halt the transmission. The stop condition is simply a voltage transition from low to high on the SDA line after a low to high transition on the SCL line, with the SCL line remaining high.

5.2 Example of a data transmission

Now that we have defined the structure of typical messages shared, below there is an example of a data transmission between master and slaves.

1. Figure 5.3: the master sends the start condition to every connected slave by switching the SDA line from a high voltage level to a low voltage level before switching the SCL line from high to low.

2. Figure 5.4: the master sends each slave the 7 or 10 bit address of the slave it wants to communicate with, along with the read/write bit.

3. Figure 5.5: each slave compares the address sent from the master to its own address. If the address matches, the slave returns an ACK bit by pulling the SDA line low for one bit. If the address from the master does not match the slave's own address, the slave leaves the SDA line high.

4. Figure 5.6: the master sends or receives the data frame.

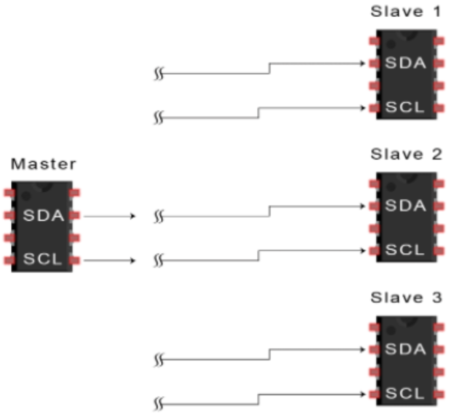

Figure 5.3. I2C communication step one

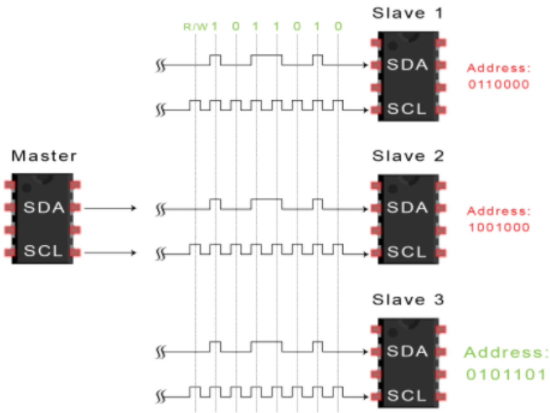

Figure 5.4. I2C communication step two

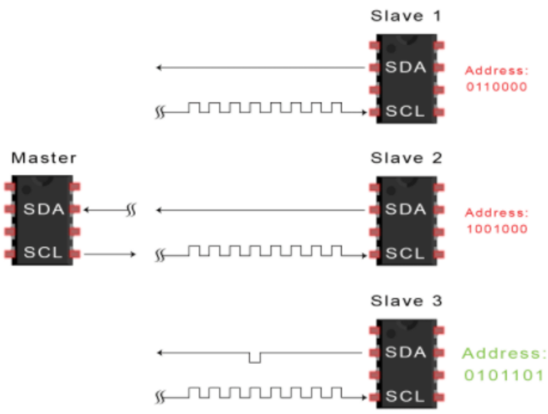

Figure 5.5. I2C communication step three

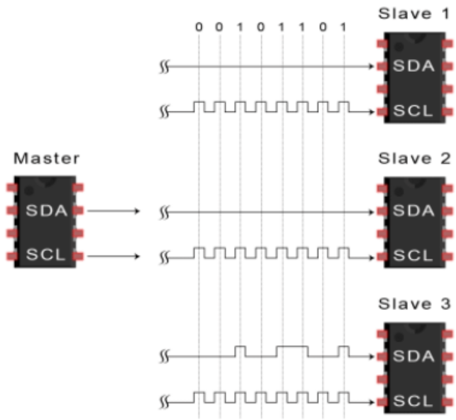

Figure 5.6. I2C communication step four

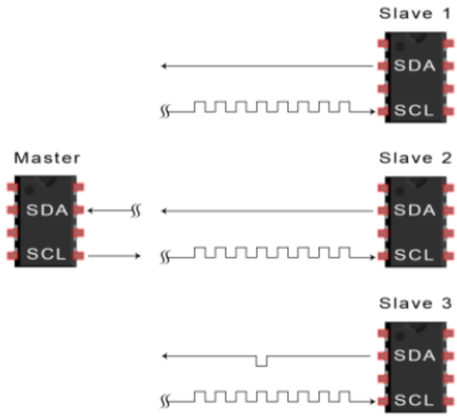

Figure 5.7. I2C communication step five

5. Figure 5.7: after each data frame has been transferred, the receiving device returns another ACK bit to the sender to acknowledge successful receipt of the frame.

6. Figure 5.8: to stop the data transmission, the master sends a stop condition to the slave by switching SCL high before switching SDA high. [15]

It has been seen that the bus is composed of two wires, meaning that communication needs only two lines. However, in practice it is essential to have a common reference wire, generically and generally called ground (GND), and a power connection line. Regarding the positive power connection, usually values of +5 V or +3.3 V are used, and it is not even essential that it is common to all the connected components, since there is also the possibility of using different power voltages for the connected devices, although in most applications the power tension is common. Moreover, at hardware level, SCL and SDA lines are open-drain or open-collector depending on the technology used if MOSFET or BJT respectively. Therefore, the presence of a pull-up resistor - Figure 5.9 - is essential. When a device activates its output, it forces down the line bringing it to logic zero level. Leaving it free it is brought to the supply voltage by

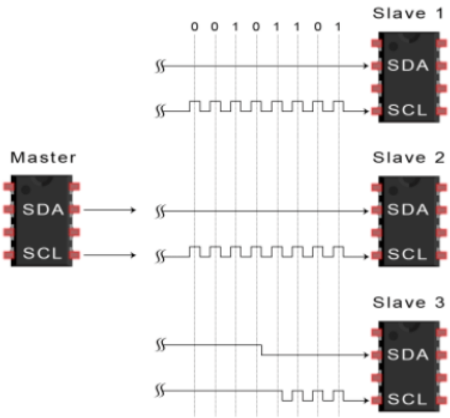

Figure 5.8. I2C communication step six

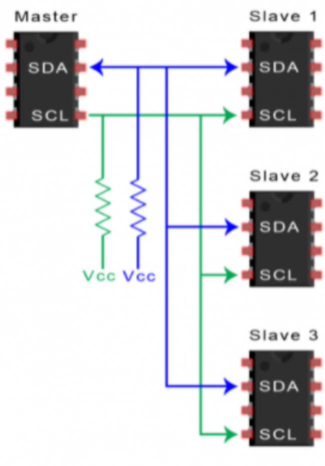

Figure 5.9. I2C scheme of the pull-up resistors connection

the pull-up resistor and is considered at logic high level. This has the advantage that there are no hardware conflicts in any case, in the sense that no device can force the logic high level. It should also be mentioned that in order to improve the reading in the presence of disturbances a hysteresis is often present on the inputs of the devices. [12]

Now that we've discussed in detail how data is exchanged inside the generic appliance, it is necessary for the continuation of the project, to understand what kind of appliances can be used for the primary frequency regulation. In the next chapter we will investigate trying to find similarities between the behavior of different appliances that may be present in a typical residential house.

Chapter 6

Home appliance classification

6.1 Purpose

The ultimate goal of this project is to transform as many households loads as possible from passive to active grid elements regarding the primary frequency regulation, even without the end user noticing it. However, not all the domestic appliances have the same operating characteristics. It is therefore necessary to recognize on which type of loads it could be possible to install a control logic that forces their switching on or off. According to this reasoning we can identify three macro-categories of household appliances.

6.2 Not suitable loads

To this category belong the household appliances usually not connected to the electrical network, except when they are to be used and which typically require human control for their correct operation. This category includes, for example, hairdryers, vacuum cleaners, shaving machines, etc. All of them in fact cannot be turned off if they are in operation due to the clear needs of the users. These appliances, therefore, cannot be taken into consideration for an active participation on the frequency regulation control.

6.3 Programmable loads

Then, there are those loads that are always connected to the network and which also have a more or less consistent absorption of electrical current. Therefore, it is possible and it would be useful to be able to have some interaction with them. They are generally managed by the user only during the start-up phase, while during the rest of the operation they are automated. In this way they are called programmable appliances. In this category we can mention the various dishwashers, washing machines and dryers. Generally, it is not possible to start them automatically when they are switched off, because the electronic board is switched off together with them. But when they are working and performing their task, it could be possible to command a forced stop for a time useful to restore the nominal parameters of the network in case the network frequency drops.

6.4 Thermal loads

The last typology of domestic loads that has been identified is the one which, like the previous one, is generally composed by appliances which are constantly connected to the electrical network, but whose functioning is not subordinated to any human action and their current absorption in time is completely managed by the electronic systems inside them. In the vast majority of cases these appliances are loads characterized by thermal inertia - such as refrigerators, boilers, air conditioners, etc - and once they reach the natural thermal steady-state imposed by the user, they could be programmed to turn off - to exploit thus their stored energy - or to turn on - to exploit their storage capacity - according to private and network needs. Doing this operation, they can be compared to traditional electrical energy storage systems such as electrochemical batteries. In this way, it would be possible to transform these inert loads into active components capable of contributing not only to the instantaneous frequency regulation but also to the programmed management of the electrical transmission network or of the domestic one. In order to do this, however, it is essential to be able to interact at the level of internal register entries with a controller that monitors the states and that even manages its activity over a period of time more or less long.

Once we have determined which appliances may be suitable to support the generation systems for primary frequency regulation, the next chapter will describe:

- the evaluations that led to the choice of hardware components for the actual implementation of the control system that we want to implement;
- the characteristics of the model loads that were chosen to conduct the laboratory tests.

Based on these choices, Chapter 8 will describe the implementation of the control logic I conducted first on the generic programmable load and then on the model chosen for laboratory testing. Similarly, in Chapter 9 the same will be done but considering the generic thermal load and the corresponding selected model.

Chapter 7

Hardware used

7.1 Logic device

In the initial phase of the project, I analyzed at which level the logic of loads control should be placed. The principle taken into account to implement this management system, as already stated, is to strengthen the primary frequency regulation of the network system, trying to increase the energy inertia of the production one, due to the consolidated and constant penetration of renewable sources, which generate large quantities of electrical power but do not provide enough inertia to guarantee easy control over the stability of the network. Thus, willing to use the domestic loads as active elements for frequency regulation, it becomes crucial both the response time and the ability to follow as quickly as possible the network status updating the sampled values with a high ratio. If we think to place the control logic at the end-user level, leaving the decision-making ability even to the single load, the problem would be to find the information about the network status in real time. A solution to this problem could be to install a frequency meter attached and connected to the load. Without the use of this meter, in fact, in order to reach this simple, although neuralgic purpose, a hardware implementation would probably be more complicated, with dedicated communication buses and certainly the introduction of delays due to communication between devices. On the contrary, if the control logic was delegated to the manager side or to a dedicated platform that can measure the electrical parameters of the grid and control the status of all the loads connected to its network, the problem becomes on how to interact with all the elements of its network. According to this scenario, in fact, a very consistent number of appliances would have to be managed. Moreover, the network to control will be continuously changing because home appliances would be added or removed in according to the requirements of the end-users, without a predictable pattern. The fundamental problem would be to let the network manager understand the type of each individual appliance registered and to design a communication protocol to communicate the actions in such a way to not interfere with the other communications intended for the different categories of loads. In this case, the hardware implementation on the appliances would be more

Table 7.1. Hypothetical scenarios for load control

Who acquires network frequency value	**Where the logic is implemented**	**Who executes the command**
Dedicated platform	Scenario 1: Dedicated platform	External switch
		Electronic board of the appliance
	Scenario 2: Device external to the appliance	External switch
		Electronic board of the appliance
	Scenario 3: Electronic board of the appliance	Electronic board of the appliance
Device external to the load	Scenario 4: Device external to the appliance	External switch
		Electronic board of the appliance
	Scenario 5: Electronic board of the appliance	Electronic board of the appliance

streamlined, having to simply receive commands and nothing else, while the software and data management side, as we can imagine, would be more complicated. As well the mathematical modeling of the home appliances would be less precise because it will be just a single one for all the different appliances produced by several manufacturers. Table 7.1 briefly summarizes the scenarios that were initially considered.

At the end of the decision process, in order to execute the start or pause command, independently from who has taken the decision, two ways are considered. In the first one a device can be installed between the power supply and the controlled load, which working as a switch can temporarily supply or de-power the appliance. The second way is absolutely more discreet but also more difficult to accomplish, because it planned to directly modify the internal register entries of the appliance or to act on its electronic board to force a start or a pause command.

- Scenario 1: the grid manager, or an appropriate platform, acquires the parameters of the electrical network, processes them, and communicates via an internet connection - probably the most convenient and efficient communication type due to the vast number of appliances involved - the command to start, pause or modify the internal register entries of each appliance according to programmable or thermically type.

- Scenario 2: for each appliance or house, there is a device connected to a platform that act as a master and communicates to all the devices the network parameters. Based on these values the single device makes a decision whether to stop or start a more or less number of loads under its control, also based on the priorities of the loads themselves.

- Scenario 3: everything is implemented in the electronic board of the appliances, which through a processor connected to the Internet acquires from a dedicated platform the parameters on the state of the network and saves them on its internal register. Therefore, each load independently decides whether to force a command to turn on or off.

- Scenario 4: there is no external platform, but a single device per household appliance equipped with a frequency meter that based on the acquired measures directly takes the decisions to be taken.

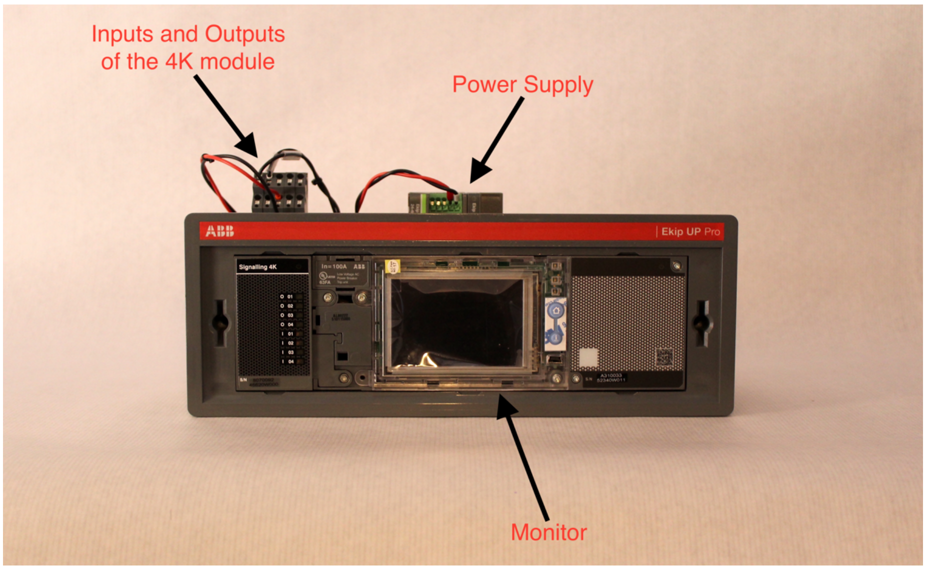

Figure 7.1. Picture of the Ekip UP Pro used as the control device

- Scenario 5: it is fully internal to the appliance, equipped with both meters of the network status and with the logic predisposed to the elaboration of the acquired data with the relative creation of dedicated internal register entries. The adoption of this solution is the most technological one but unfortunately it is for sure more accessible to the single appliances manufacturers than to us.

Considering the various pros and cons of each hypothesized scenario and the technological means at one's disposal, I have concentrated on the development of a device to be interposed between the electrical network and the appliance to be controlled as in the fourth scenario. On this device, a control logic will be uploaded and taking into account the on/off status of the load to which it will be associated and in relation to the network frequency that it directly measures, it will act on the start or pause command directly at the electronic board level of the appliance. The device that has been chosen operationally to use in this first phase of the study is the "Ekip UP Pro" by Abb [16]. It is a digital unit that allows to monitor and control low voltage plants. In Figure 7.1 is shown its picture. The reasons that conducted to its choice are:

- it is capable of measuring any type of electrical measurement;
- it is possible to program a block logic, developed by Abb and particularly useful for the implementation of a control of this type;
- it is equipped with an installable module for the processing of digital and analog signals both inputs and outputs.

Thanks to this last feature, the command signal will be communicated directly to the appliance. It simply consists of a controlled contact that in according to the implemented logic inside the device, can open or close itself.

Figure 7.2. Picture of the modified capacitor

7.2 Programmable load

For laboratory testing, as a model of all programmable loads, we chose the latest generation tumble dryer model T8DEC94ST 8000 Series 9 Kg produced by AEG [17]. A tumble dryer was chosen because it is easier to install and easier to manage during testing than a washing machine or any other appliance with water processing. In order to interface with the electronic board of this appliance, instead of adding a new element in its structure, due to the problems mentioned in Chapter 5 on I2C communication, it was decided to trick the microprocessor associated with the physical start/pause button next to the HMI of the dryer, simulating a pressure every time we want to change the state of the load. The start/pause key in question that we want to modify for automatic dryer control is a capacitive type button. The micro-controller of the key, at its physical pressure perceives a change in capacity, going to communicate to the microprocessor that the key has been pressed. To control it remotely, instead of intervening between controller and processor and then going to change the register entries, it is better to trick the micro-controller, making it vary the capacity that it measures, although physically there is no pressure. To do this, on the electronic board, in correspondence of the physical button a new capacitor is built - Figure 7.2 -. It is composed by the real physical key - the first armature -, a strip of dielectric material - insulating material placed between the two armatures - and a copper strip - the second armature -. The two armatures of this new capacitor, which is now the one seen by the micro-controller, may or may not be electronically short-circuited by a new custom-built circuit board. When short-circuited, the capacitor that the controller would look at would be the one made up of the two faces and the outside world. By doing so, the custom board - Figure 7.3 - can simulate the pressure of the

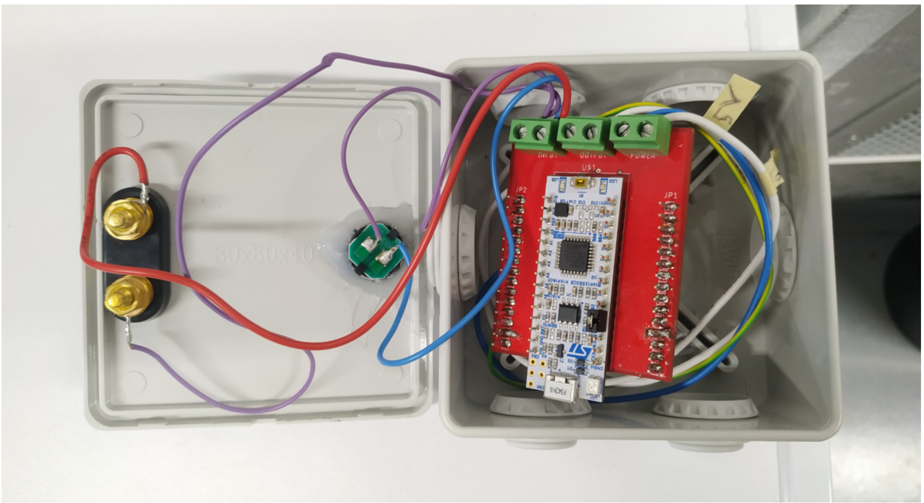

Figure 7.3. Picture of the electronic board designed for the dryer control

physical button, changing the structure of the capacitor and thus making it vary the capacitance measured by the controller, which is thus deceived.

This new fundamental component for the success of our control has been designed and built by the engineer Andrea Lopresti and the engineer Alessandro Mansutti of the Rold S.r.l. company, subsequently a meeting in which the needs and the detailed operation of the Ekip UP Pro device have been explained in detail. Constructively speaking, in installation, the custom board is connected to the positive pole and to the ground of the electronics of the household appliance so that it can be powered. Then it has obviously connections to the two faces of the capacitor previously described. To control the dryer, the custom board also has two pins that when they are put in contact, via Ekip UP Pro output signal, the custom board understands that it is time to make the capacitance change. So, when these two contacts are short-circuited, the board knows that it must change the state of the capacitor and the appliance will pause or resume autonomously. Operationally there is a start/stop command only when there is the transient in which the output contacts of the 4K module of the Ekip UP Pro switch from open to closed position. The reverse, in fact, does not cause any change of state in the capacitor of the capacitive key. Therefore, assuming a scenario in which we want to start, pause and then resume the program of the dryer, the commutations, starting from an open state of the contacts, should be:

$I \rightarrow O \rightarrow I \rightarrow O \rightarrow I \rightarrow O$.

7.3 Thermal load

As a model for all thermal loads we have chosen to use an oven model FA3S 844 IX HA by Hotpoint/Ariston [18]. This type of appliance has been chosen because it is one of the most energy-consuming in this category and because when it is on and working it fully represents the behavior of its class. For its electronic control, an

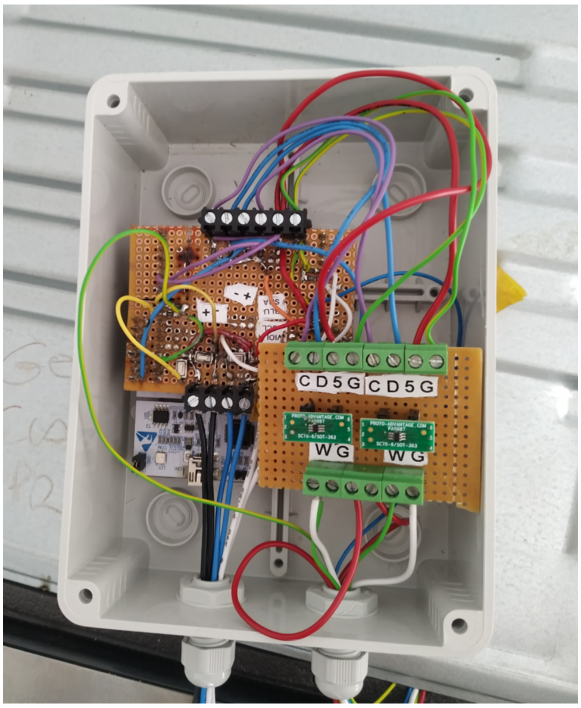

Figure 7.4. Picture of the electronic board designed for the oven control

interface has been created by modifying the two physical knobs of program selection and temperature selection, in such a way as to be able to vary their status no longer by turning them manually, but by short-circuiting the ends of cables connected to a custom-built electronic board as done for the dryer. For this hardware modification of the home appliance we have once again relied on the skills of the engineer Andrea Lopresti and the engineer Alessandro Mansutti of the company ROLD S.r.l., who have developed an electronic board - shown in Figure 7.4 - that could act as a logical link between the home appliance and the control device - the Ekip UP Pro -.

As stated, the general idea for its control is practically the same as that developed for the dryer. Constructively speaking, for oven control, this new custom board has two pairs of pins at the output. By short-circuiting one or the other - via the output of the 4K module, similarly to what was thought up for the dryer - it's like turning the temperature knob clockwise or counterclockwise, respectively. Unlike the dryer, therefore, a change in output from open to closed, does not determine the pause or resumption of the appliance, but is only equivalent to a click of the knob that varies the temperature set by the user of $\pm$ 20° according to the pin that is short-circuited.

Chapter 8

Custom logic for programmable loads

As declared in the previous chapter, the status of the controlled load is commanded by the Ekip UP Pro through the 4K module which is possible to install in it. To govern the status of its output contact, which are the one connected to the custom board capable of pausing and resuming the appliance, it was decided to use a logic available in the ABB products. The command that forces a pause in the drying cycle when the mains frequency falls below a certain threshold and forces its resumption once the frequency returns above the threshold, will be managed by the Custom Logic. It is a programming language programmable via the Ekip Connect software. It is nothing more than a logic block language that can combine different blocks with each other and with various inputs, outputs and states of the device, to obtain the desired result. The following will illustrate the evolution and all the steps that led to the final form of the Custom Logic that controls the dryer considered as a model for all the programmable loads.

8.1 Ekip UP Pro configuration

In order to start programming Custom Logic, it is necessary to have the Ekip Connect program and the Ekip TP Abb port [19] for the PC/Ekip UP Pro connection. After connecting through the TP port the device to the PC and launching the Ekip Connect program - in this project I am using its version 3.2.9.13 - we have to perform a scan to find the device and be logged as Internal RD of the Abb staff to take advantage of the options in its Reserved Menu. Starting from a new Ekip UP Pro, without any configuration yet, to enable the Custom Logic in the device, it is necessary to activate it from the “Reserved Menu” of Ekip Connect, selecting “Prot Enabled By Par” in the “Reserved Unit config” section as shown in the snapshot 8.1. After this step, check “PLC Custom Logic” - Figure 8.2 -. At this point we can enable from “Programmable Status and Outputs”, the tick “PLC Enable” from the “Custom Logic” menu that has now appeared - Figure 8.3 -. To avoid in the loading of the Custom Logic on the

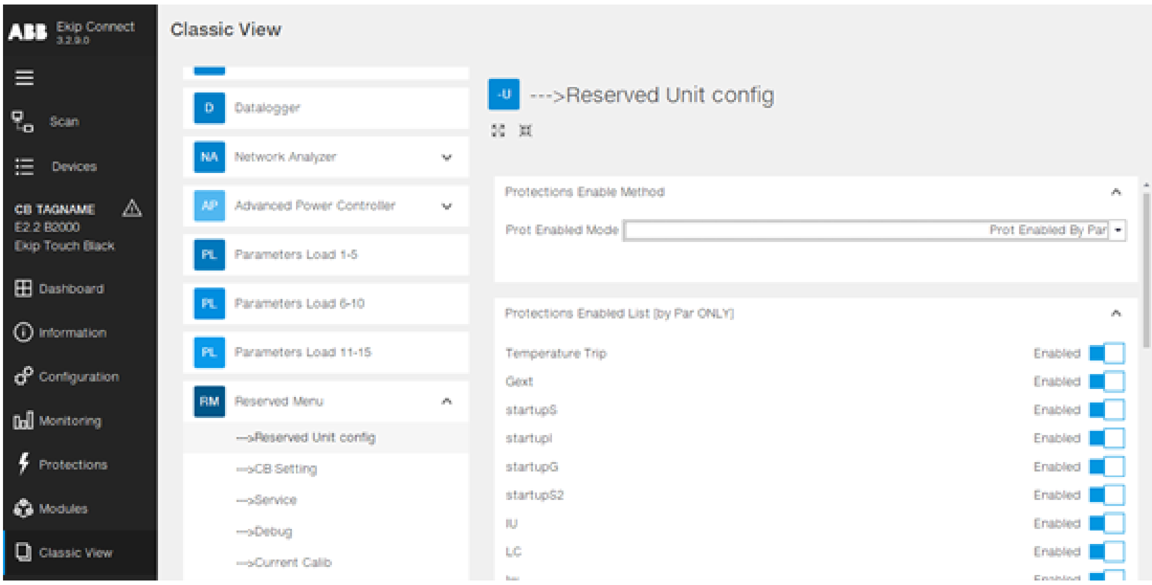

Figure 8.1. Selection of “Prot Enabled By Par”

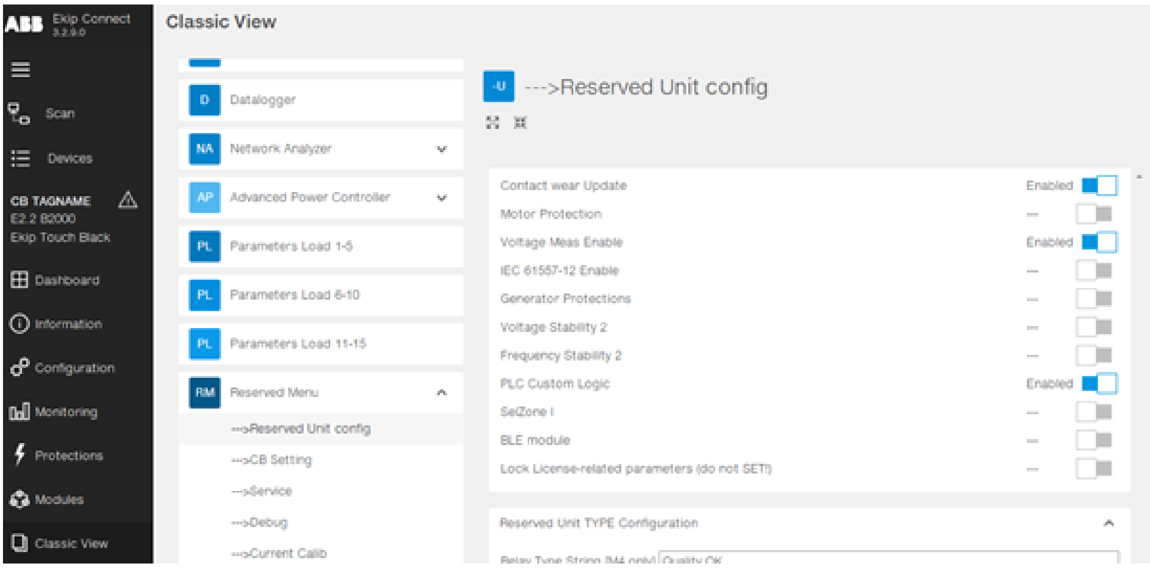

Figure 8.2. Check “PLC Custom Logic”

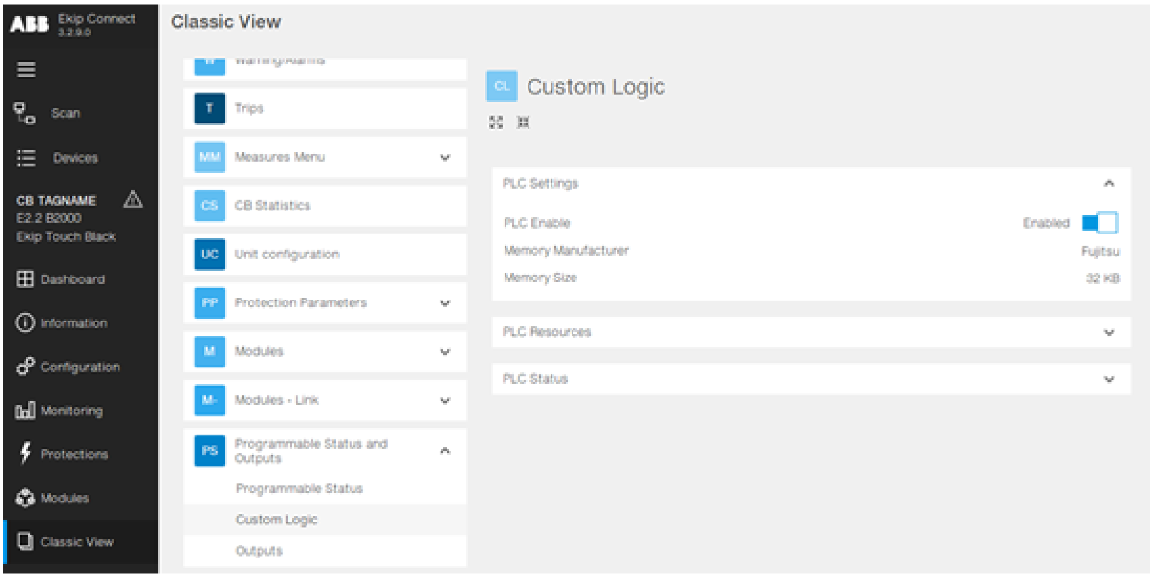

Figure 8.3. Tick "PLC Enable" from the "Custom Logic" menu

device an error message due to the absence of a measurement module, again from "Reserved Menu", under "Reserved Unit Config", it is necessary to activate the button "NONE" in "None protection config". After this passage, press "Ekip up protect" button under "Ekip up protection config" menu.

After these operations it is now possible to open the "Tools" menu and start to program the Custom Logic.

8.2 First attempt

Initially I did not examine the real behavior of the dryer, trying rather to program something universal, and then only later make the necessary changes to adapt the control logic on the specific case of our real appliance. As a first step, being essential the use of the 4K module, if it should not be present, it would be necessary to communicate immediately an error - Figure 8.4 -. Therefore, an input on the presence of this module ("Ekip Signalling 4K present") has been denied, represented by the empty dot instead of the arrow, and associated to the first output of the PLC ("Errore del 4K"), which in case of high signal, it will have to display an error message. One of the problems that may occur is that the load may start untimely if it is turned on while the 4K output control is sending a signal. It is also useless for the 4K module to send signals if the appliance is off or not powered. To overcome these two problems, in order to understand when the load is effectively on, as shown in the figure 8.5, it would be useful to use an input of the 4K module ("Ekip Signalling 4K.1 IN"), where a DELAY ON of 3 seconds is added to the logic high state. Thus, any command sending can only take place if more than 3 seconds have passed since the load itself was turned on ("accensione da ferma" or "power on from standstill"). To do this, therefore, when modifying the load, it would be necessary to connect an

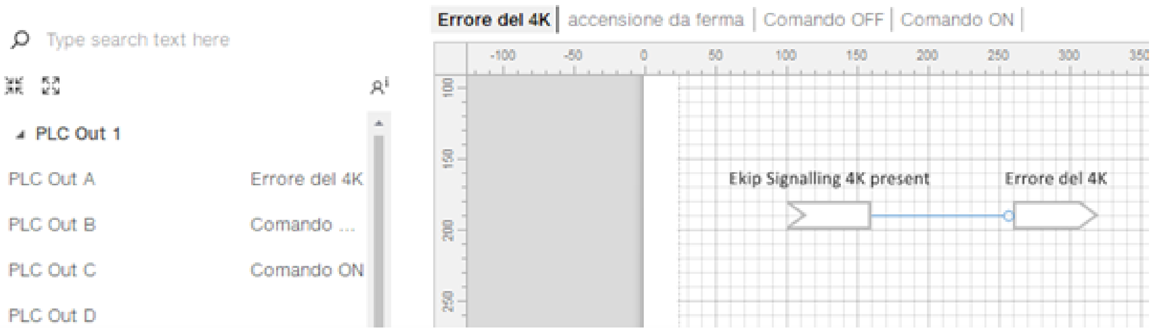

Figure 8.4. First attempt of the rule in case an error in the 4K module occurs

Figure 8.5. First attempt of the rule that detect if the load is powered on

input contact of the 4K module - the 4K.1 IN - to the electronic board of the load in order to understand if it is on or off - or find another way to declare it -. A priori, however, it is not possible to determine whether operationally the absence of this control could cause malfunctions. At this stage, however, I chose to include it in the logic. Thinking about normal operation, after the load has started up, the control over the mains frequency should come into operation. When the frequency drops below a minimum threshold, the output of the 4K module connected to the custom board must close and reopen to give the signal to trip the start/pause button. It is necessary to give this command when the network frequency falls below the minimum threshold ("Frequency warning 1") and only if the appliance controlled is turned on ("accensione da ferma" or "on from standstill") and started, otherwise since it is paused it starts – and this is just the opposite of the willing behavior -. At this stage, it is not yet determined how - it may be that a current or power threshold has been exceeded - but the "programmable status A" ("è in asciugatura?" or "is it drying?") represents the start/pause operating status. If all input conditions at the AND port are verified, a high signal with a set duration of 200 ms is generated by the PULSE block - Figure 8.6. This pulse will be the final output of the pause command ("Command OFF") that will be associated with the closing of the output of the 4K module connected to the custom board. However, this control logic is still too rudimentary. For example, we need to make sure that the network frequency does not just fluctuate around the threshold value, but remains persistently below it, especially since constructively the

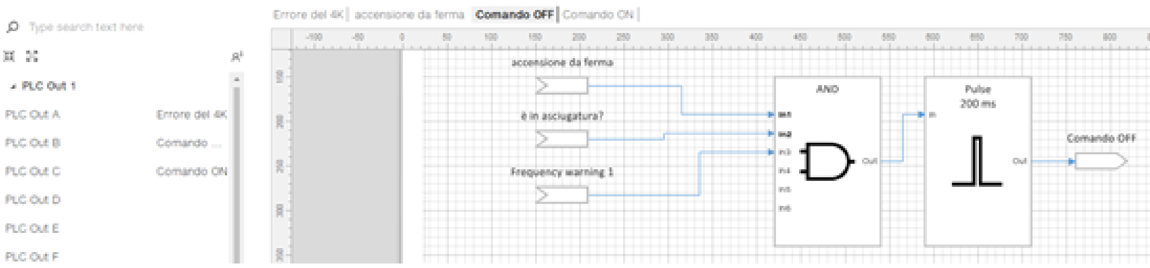

Figure 8.6. First attempt of the command off rule

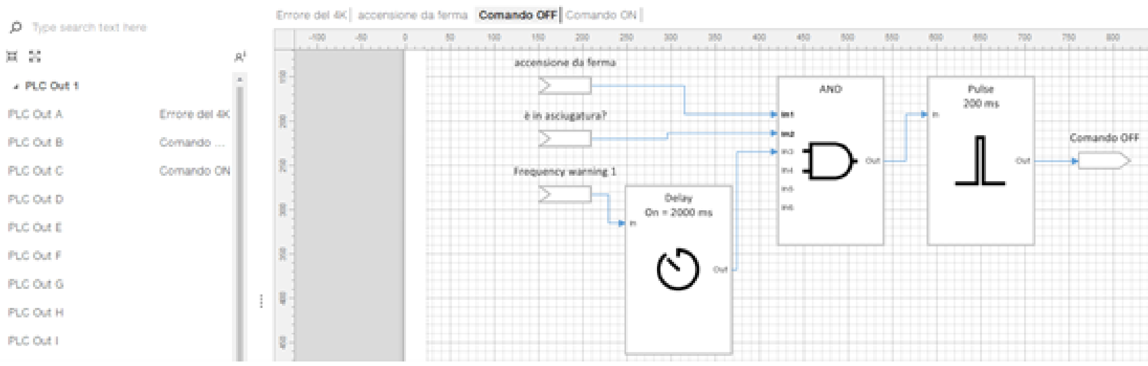

Figure 8.7. Modification to the first attempt of the command off rule

custom board has a delay of about 1 second between the closed operation of the 4K associated output and the stop signal it sends to the load. So, let's assume that the network frequency needs to stay below the threshold for at least 2 seconds. Therefore, I introduce after the input of "Frequency warning 1" a DELAY ON equal to this value. The modifies are visualized in the figure 8.7. Furthermore, in order not to have situations where the operation is interrupted and restarted too often in a short period of time - so as not to compromise the efficiency of the work done by the load - it is established that at each resumption signal, the load must be working for at least 5 minutes before being paused again if necessary ("Ekip Signalling 4K.1 OUT"). Since the resumption command occurs when the output of the 4K is closed, a DELAY OFF denial is set to debilitate the AND port until this time has passed since the last command was sent. This second modification is represented in the figure 8.8. In order to program the resumption command, which is shown in the figure 8.9, it must come into operation only if the load is on ("accensione da ferma" or "on from standstill"), not running ("è in asciugatura?" or "is it drying?") and previously the pause command ("Command OFF") has come into operation. This last condition must be taken into account, otherwise there may be a risk that the load is started untimely just because the mains frequency is above the minimum threshold. The AND port will therefore be enabled only after the pause command has been given, keeping this signal high by means of a DELAY OFF for a sufficient time for the mains frequency to return close to the nominal value, which for the moment I exaggerate to be 24 hours. After the load has been paused by frequency control, it can only be restarted again when the mains frequency returns above the minimum threshold and

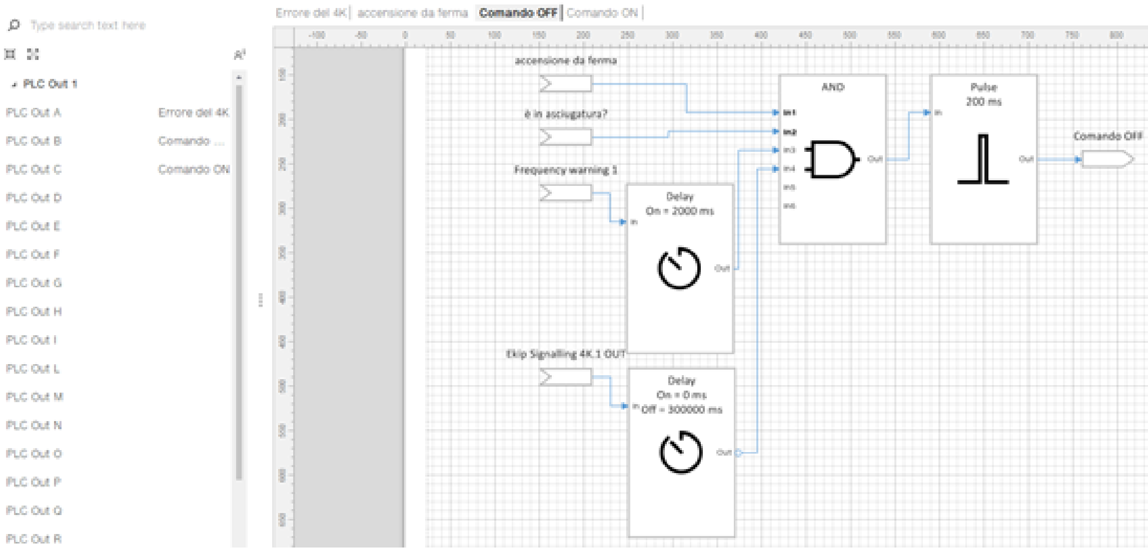

Figure 8.8. Second modification to the first attempt of the command off rule

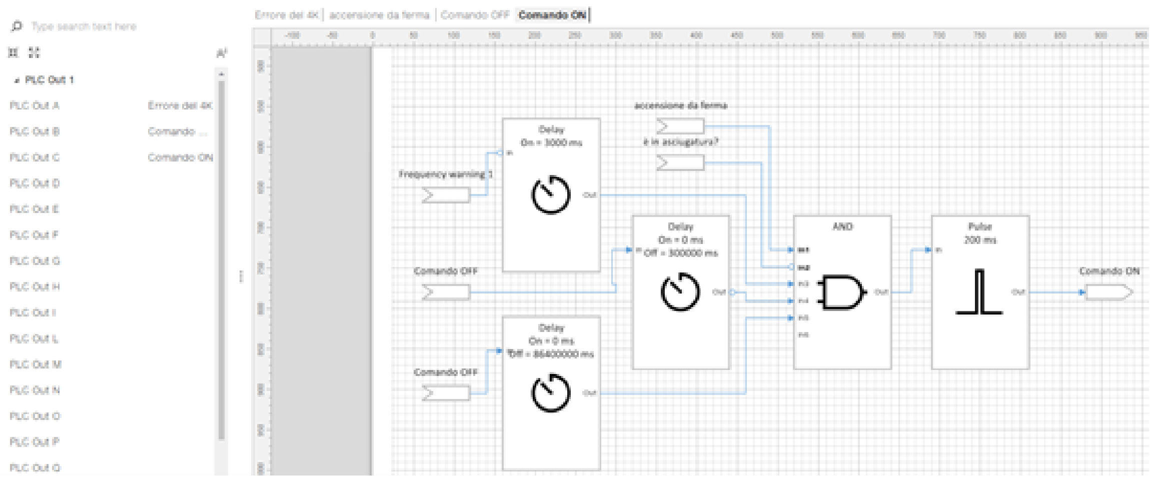

Figure 8.9. First attempt of the command on rule

remains there for at least 3 seconds. This is achieved by negating "Frequency warning 1" and adding a DELAY ON. In order to avoid that the signal is ignored by the dryer in case it comes immediately after the pause signal - due to the 1 second execution delay of the custom board -, even if intrinsically it is already safe because of the delays set on the frequency alarm status changes, and to avoid too short stop and restart transients, a minimum time is set with a DELAY before sending the signal. After it has given the pause command ("Command OFF"), in fact, it keeps this signal high long enough with a DELAY OFF denied, so that it debilitates the AND operator and avoids sending new commands. If it has just sent the pause command, a minimum waiting time of 5 minutes is therefore set before it resume again. At the output of the AND port, in order to generate a signal compatible with the operation of the custom board, as for the pause command, a PULSE of 200 ms duration is set ("Command ON"). Then, for the test, the outputs "PLC Out B" ("Command OFF") and "PLC

Out C" ("Command ON") should be set so that if one of them is high, then the output of the 4K.1 OUT should be closed - this is how we tell the custom board to act on the capacity of the start/pause button -. Note: the times set at this stage are completely arbitrary, only later I will study what times are best to set for optimal operation of the load on which this system is to be installed.

8.3 Debug

Since this is the first attempt at creation, let's debug briefly and analyze which conditions deny sending the start/pause commands. For the "Command OFF", the AND port returns a high value when simultaneously answering the following questions in this way:

1. Has it been at least 3 seconds since power-up? YES
2. Is it drying? YES
3. Is the grid frequency lower than FW1 threshold? YES
4. Was a command given with the 4K module within 5 minutes? NO

For "Command ON", the AND port returns a high value when the following questions are simultaneously answered in this way:

1. Has it been at least 3 seconds since power-up? YES
2. Is it drying? NO
3. Is the grid frequency lower than FW1 threshold? NO
4. Has OFF Command been activated within 5 minutes? NO
5. Has OFF Command been activated in previous 24 hours? YES

Imagining that the tumble dryer starts from a power-off condition, the behavior described in the table 8.1 is compiled. The numbers indicate the questions that deny sending the command.

Thinking about programmable loads, for a single operation cycle everything is regular and the logic seems to work as established.

However, let's suppose that in the middle of the operation the load is turned off and on manually by the user for a very short time, as it happens when someone realizes that a mistake was made in the setting operations program or in case of a black out, and try to recompile the table 8.2 assuming that in the previous 24 hours a command off was sent.

If the load has been turned off and on again within 24 hours since the last time the off command was sent, then it may start up for the first time on its own untimely.

Table 8.1. Debug table of the first attempt of the programmed Custom Logic

	Command OFF	**Command ON**
powered off		
f $\geq FW1$	Nothing happens: 1, 2, 3	Nothing happens: 1, 5
f $\leq FW1$	Nothing happens: 1, 2	Nothing happens: 1, 3, 5
Powered on and passes less than 3 s		
f $\geq FW1$	Nothing happens: 1, 2, 3	Nothing happens: 1, 5
f $\leq FW1$	Nothing happens: 1, 2	Nothing happens: 1, 3, 5
Powered on and passes less than 5 min		
f $\geq FW1$	Nothing happens: 2, 3	Nothing happens: 5
f $\leq FW1$	Nothing happens: 2	Nothing happens: 3, 5
Powered on and passes more than 5 min		
f $\geq FW1$	Nothing happens: 2, 3	Nothing happens: 5
f $\leq FW1$	Nothing happens: 2	Nothing happens: 3, 5
The load starts working and passes less than 5 min		
f $\geq FW1$	Nothing happens: 3, 4	Nothing happens: 2, 5
f $\leq FW1$	Nothing happens: 4	Nothing happens: 2, 3, 5
The load is working and passes more than 5 min		
f $\geq FW1$	Nothing happens: 3	Nothing happens: 2, 5
f $\leq FW1$	Sending command off	Nothing happens: 2, 3, 5
Load paused by the command off and passes less than 5 min		
f $\geq FW1$	Nothing happens: 2, 3, 4	Nothing happens: 4
f $\leq FW1$	Nothing happens: 2, 4	Nothing happens: 3, 4
Powered on and passes more than 5 min		
f $\geq FW1$	Nothing happens: 2, 3	Sending command on
f $\leq FW1$	Nothing happens: 2	Nothing happens: 3
Resume work by command on and passes less than 5 min		
f $\geq FW1$	Nothing happens: 3, 4	Nothing happens: 2
f $\leq FW1$	Nothing happens: 4	Nothing happens: 2, 3
The load is working and passes more than 5 min		
f $\geq FW1$	Nothing happens: 3	Nothing happens: 2
f $\leq FW1$	Sending command off	Nothing happens: 2, 3
behavior is repeated identically to the above		

Table 8.2. Debug table of the first attempt of the programmed Custom Logic in case of a previous command off

	Command OFF	**Command ON**
powered off		
f $\geq FW1$	Nothing happens: 1, 2, 3	Nothing happens: 1
f $\leq FW1$	Nothing happens: 1, 2	Nothing happens: 1, 3
Powered on and passes less than 3 s		
f $\geq FW1$	Nothing happens: 1, 2, 3	Nothing happens: 1
f $\leq FW1$	Nothing happens: 1, 2	Nothing happens: 1, 3
Powered on and passes less than 5 min		
f $\geq FW1$	Nothing happens: 2, 3	Sending command on UNTIMELY
f $\leq FW1$	Nothing happens: 2	Nothing happens: 3
The loads starts and then the behaviour is the same as the table 8.1		

In the same way, if the load is running and the off command has come into operation at least once, if the load were manually paused, a resume command would be sent immediately. In this case, however, because of the way the system is built, this signal could probably be ignored because it is almost contemporary with the manual pause command sent by the user itself.

Since I have noticed the criticalities of “Command ON” after “Command OFF” has been activated, let’s suppose that this command has already been activated and before the set time that enables the AND door expires, various events occur. I will use the example of the tumble dryer model that will be used for the practical tests:

1. The dryer is running, the power goes out and then comes back on after a short period of time – typical behaviour of a short black out -.
2. Tumble dryer is running and manually paused.
3. The dryer has finished the drying cycle naturally.

For the different scenarios it happens that:

1. The load restarts itself (it is wrong if there is a program reset, otherwise it would be correct).
2. A restart command is sent but it is ignored because it is sent at the same time as the manual command (correct).
3. A restart command is sent (to be evaluated if it is considered or ignored) with the risk to restart the program from the beginning.

Let’s suppose the scenarios are the same but this time the frequency protection check has not been activated or the minimum time for the AND port to enter the restart command has expired:

1. It does not restart and remains paused (it is correct if there is a reset, otherwise wrong).
2. Nothing happens and remains paused (correct).
3. Remains stationary (correct).

Previously I mentioned that we do not know how useful it is to install a probe that evaluates the switching on or off of the load electronic board. Posing now the same scenarios but removing the logic that evaluates the on or off state of the dryer to simplify future hardware installation, the behavior changes only for the first scenario. After the protection command on the mains frequency has come into operation:

1. The command is sent but the tumble dryer is off and it is ignored. When it then turns back on, it remains paused. If it happens that the power goes out and comes back while it is stopped because of the protection control, after 5 minutes it will however restart.

While in case the OFF command has not been activated:

1. It does not restart and remains in pause.

In the specific case of our model of dryer, after it has been started, if the plug is unplugged and plugged in again, the electronic board maintains the settings and information on the status of the cycle that was running but it remains paused, so it must be resumed manually. It is therefore necessary to signal to the user that the power has gone out and that it may be necessary to manually restart the dryer, because it would not do so on its own. Otherwise, a possible solution to contrast the effects of a black out could be to design a new logic rule that when the power goes out, once re-powered it will resume the drying cycle from where it left off. However, this would certainly involve the use of probes to measure the supply voltage.

8.4 Test n.1

To check if the programming done so far on Custom Logic works, some tests are performed with the newly modified Ekip UP Pro, simulating the behavior of the dryer. For convenience and to make it work better, I make some changes - shown in the figures 8.10 and 8.11 - to the logic of the basic program previously described:

- Replace “Programmable status A” with “CB closed”, which defines the running condition or pause state of the dryer.
- When the 4K.1 output signal turns on, manually close or open the Circuit Breaker (CB), simulating the start or pause of the drying cycle.
- Replace “Ekip Signalling 4K.1 IN” with “Ekip signalling 4K present”.

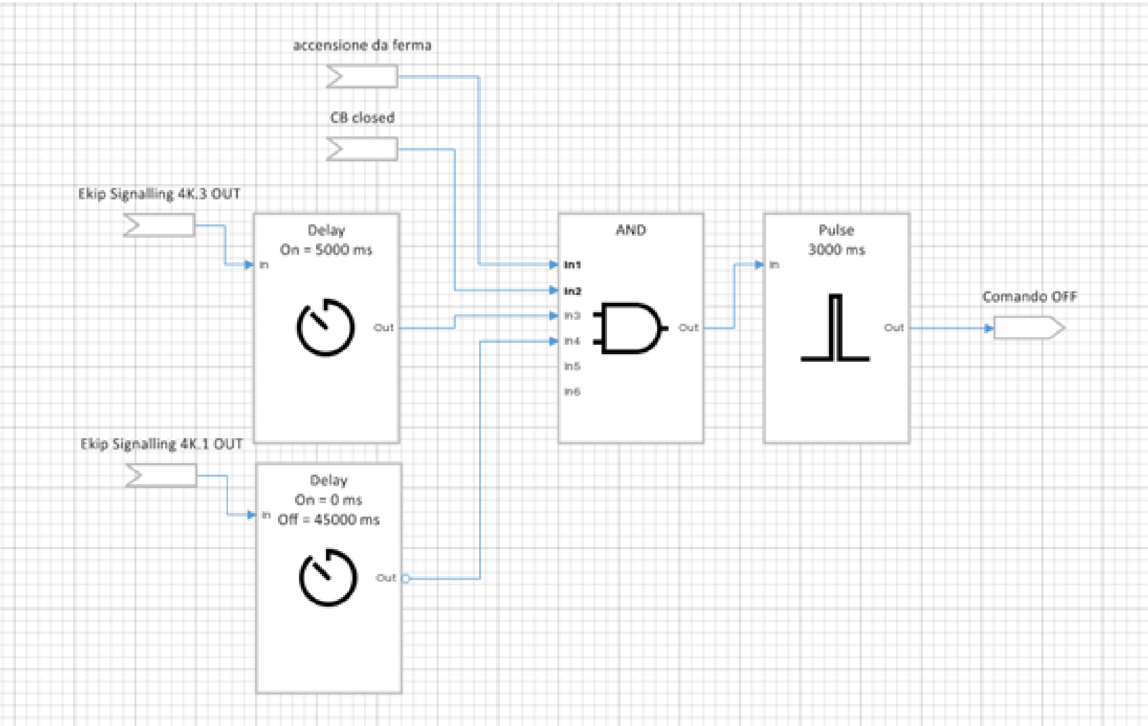

Figure 8.10. Command off rule for test n.1

- Make commands audible by increasing PULSE time to 3 seconds.
- Decrease DELAY OFF time from 5 minutes to 45 seconds and from 24 hours to 2 minutes.
- Replace "Frequency Warning 1" input with "Ekip Signalling 4K.3 OUT" and increase associated DELAY ON time to 5 seconds. Manually determine when the frequency drops below the threshold by turning on from the Ekip Connect software this output.

After editing the above, program from Ekip Connect as follows:

- Make "4K.1 Out" ON as shown in the snapshot 8.12, when "PLC out B" or "PLC out C" are true (at this stage they say whether to open or close the CB, i.e. start or pause the dryer).
- Make "4K.2 Out" ON, as in Figure 8.13, when "PLC Out A" is true (4K module error).

Proceeding to a test in which I simulate the occurrences of frequency dips and rises according to the debug table relied upon in the previous paragraph, the behavior is exactly as expected in it, with no malfunctions or conflicts in the logic.

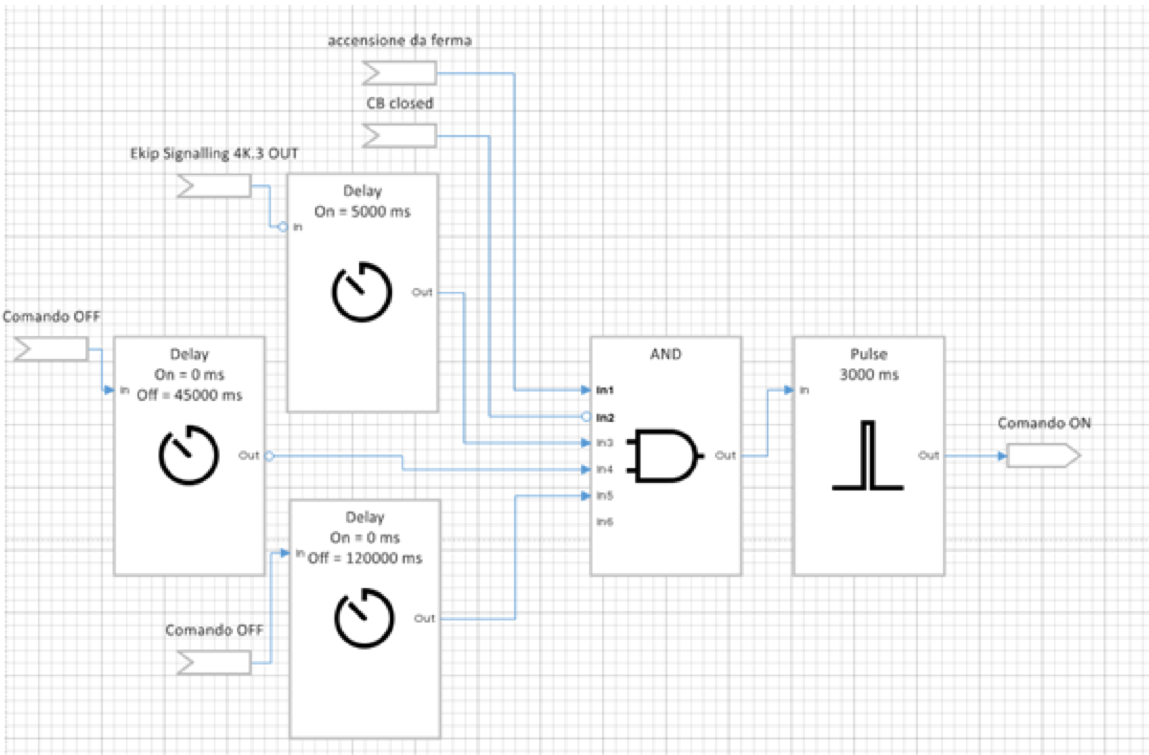

Figure 8.11. Command on rule for test n.1

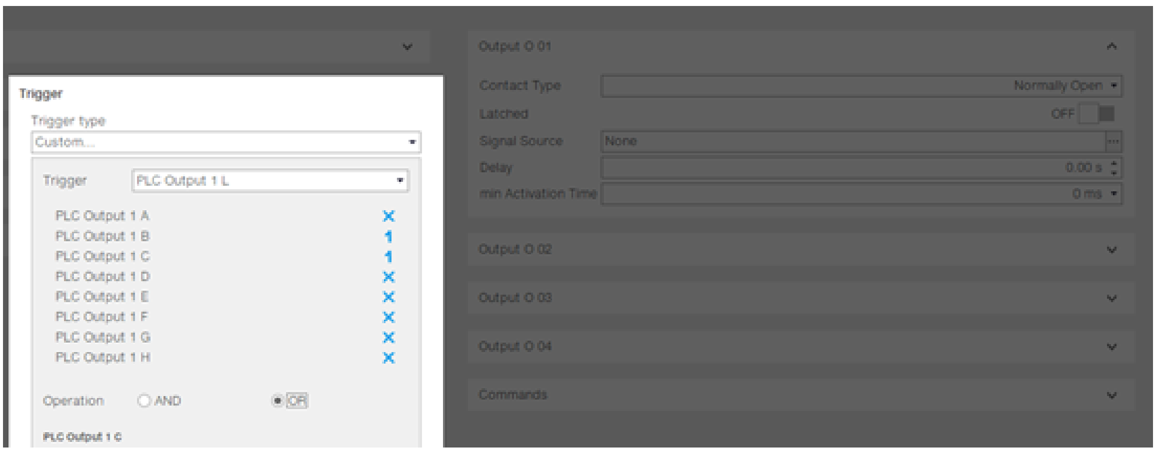

Figure 8.12. Assignment of the output command for test n.1

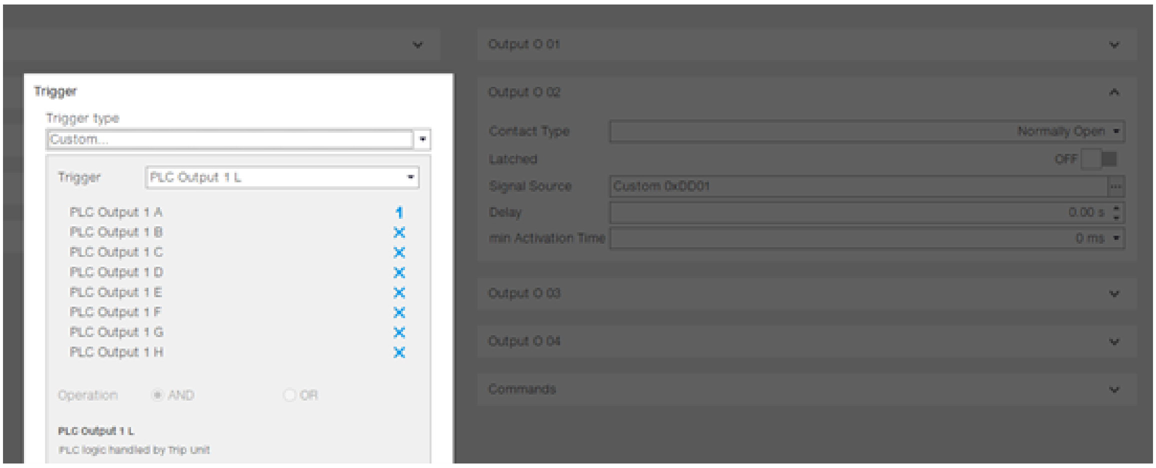

Figure 8.13. Assignment of the 4K module error for test n.1

8.5 Test n.2

For this second test, through the use of an Abb Totem, which is shown in the picture 8.14, where there is a digital simulation program in which it is possible to vary the parameters of the power supply network, I tested the operation of the Custom Logic simulating the variation of the network frequency.

In the Ekip Connect program, it is now necessary to establish the frequency value below which the “Frequency warning 1” alarm is activated. As shown in Figure 8.15, under “Other Settings A” of “Protection Parameters”, set the underfrequency alarm threshold FW1 to 0.95 Fn. Since the nominal frequency is 50 Hz, the threshold value is 47.5 Hz.

As with test n.1, assign the output of the 4K.1 Out to turn on if the status of “PLC Out B” or “PLC Out C” is high - Figure 8.16 -. When this output turns on, you will manually need to change the status of the CB, simulating starting or pausing the dryer.

As shown in Figure 8.17, assign the output of the 4K.2 Out to be turned on in the case of “PLC Out A” high.

Assign the output of the 4K.3 Out to be turned on - Figure 8.18 - in case the FW1 alarm of the previously set frequency drop is present, to detect when effectively the alarm trips.

In the Custom Logic, keeping the changes made for the test n.1, it is only necessary to re-substitute in the input to the AND ports the block “Frequency warning 1” instead of “Ekip Signalling 4K.3 OUT”. Conducting this test, as in the test n.1 the behavior is as desired, although there is a hysteresis effect when the underfrequency warning FW1 is to be deactivated. Once the frequency drops below the set threshold, the alarm is activated as programmed. However, it is not deactivated when the frequency returns above this threshold, but above the threshold plus a factor of about 0.03 the nominal frequency. This hysteresis phenomenon can be reduced or eliminated in the future by building a device dedicated exclusively to this type of control on household

Figure 8.14. Picture of the Abb Totem

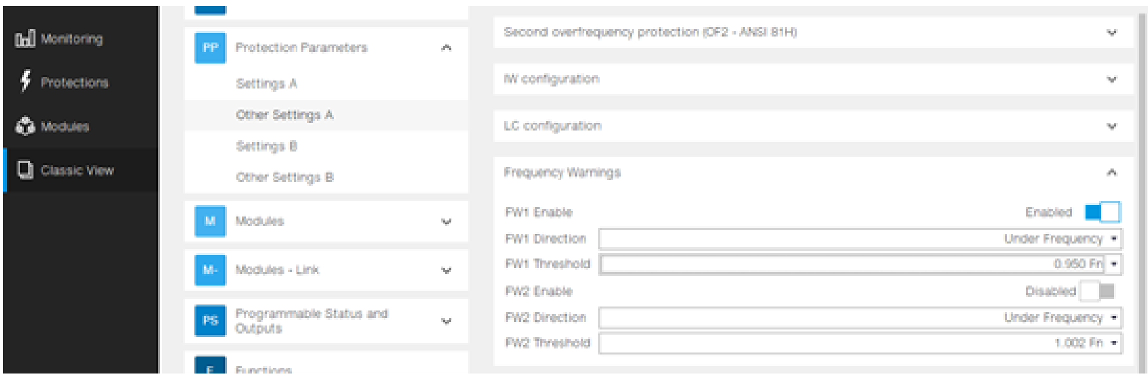

Figure 8.15. Set frequency alarm FW1 for test n.2

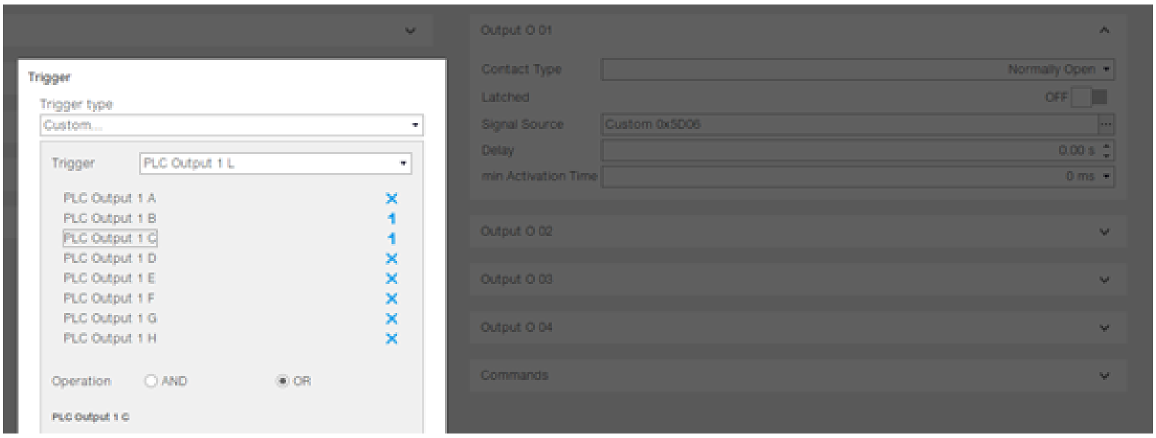

Figure 8.16. Assignment of the output command for test n.2

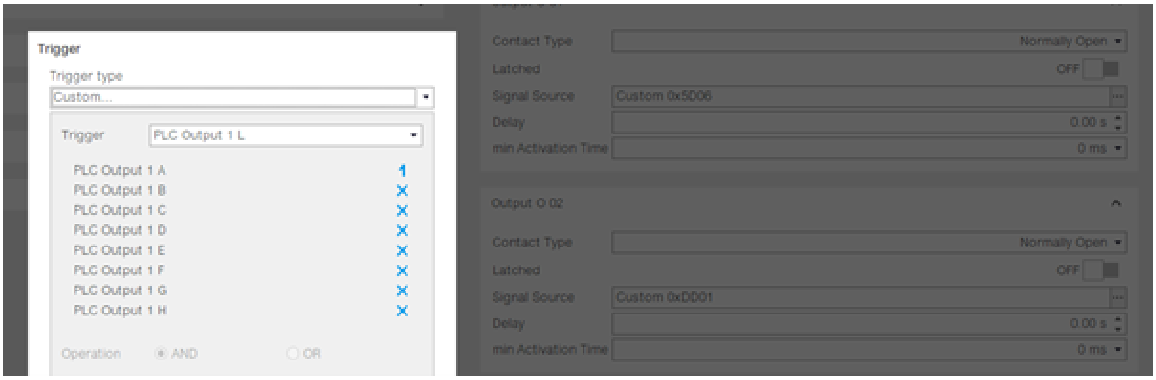

Figure 8.17. Assignment of the 4K module error for test n.2

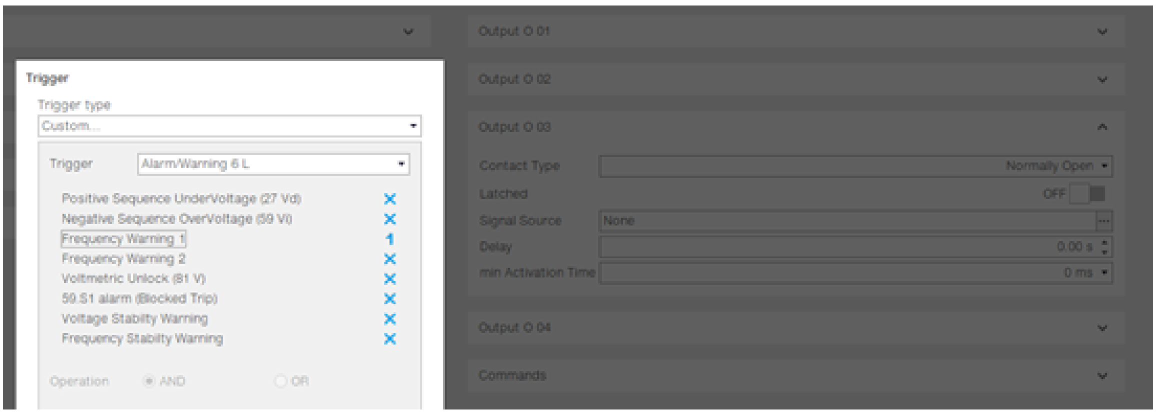

Figure 8.18. Assignment for the alarm FW1 trip detection

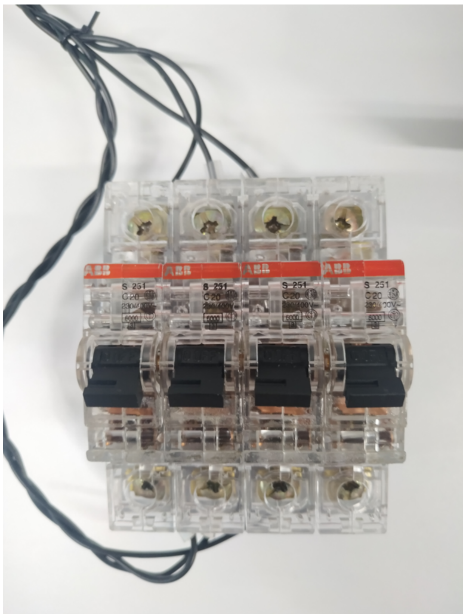

Figure 8.19. Picture of four switches, three of which were used for test n.3

appliances.

8.6 Test n.3

In this test for the first time I started using the dryer - as previously mentioned the model T8DEC94ST 8000 Series 9 Kg model by AEG - chosen as the model for all programmable loads. Cause it is the first test on the physical dryer, it was fully simulated. It was connected to three inputs of the 4K module of the Ekip UP Pro, three switches that manually can be opened or closed - Figure 8.19 -. By controlling them manually following the actual states of the dryer will simulate through these:

- Its power on or off (4K.1 IN).
- The start or pause state of the tumble dryer (4K.2 IN).
- The frequency alarm FW1 (4K.3 IN).

It was chosen at first to simulate its states so as to minimize hardware changes to be made later to the components and to the dryer itself. At the 4K.1 OUT output I connected the contacts of the custom board of the dryer through two bushings. For connections starting at the 4K module of the Ekip UP Pro, refer to the images 8.20 and 8.23. [20]

In this test, a few changes are made at the Custom Logic level. Taking as reference the modifications already made in Test 1:

- Modify the PULSE making them equal to 0.5 s.

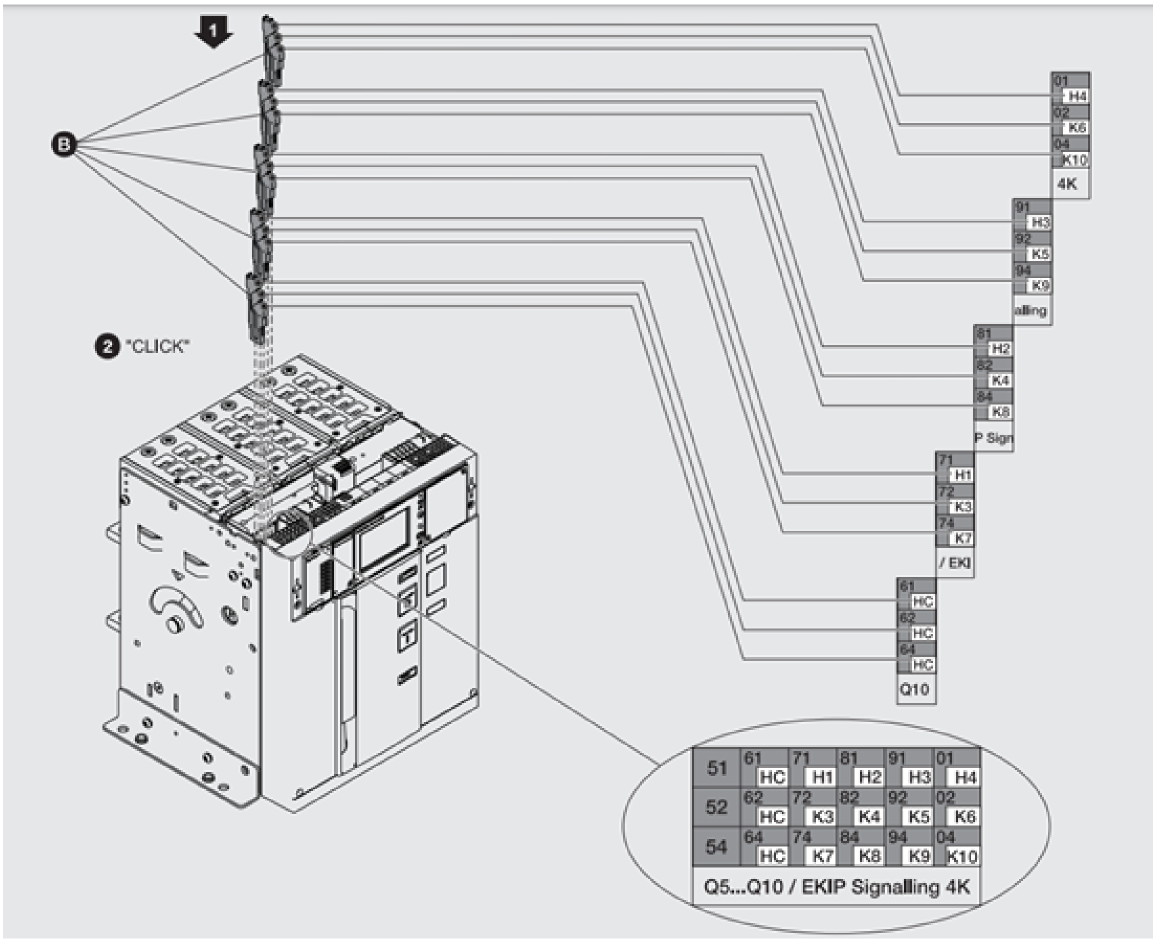

Figure 8.20. Hardware view of the 4K module

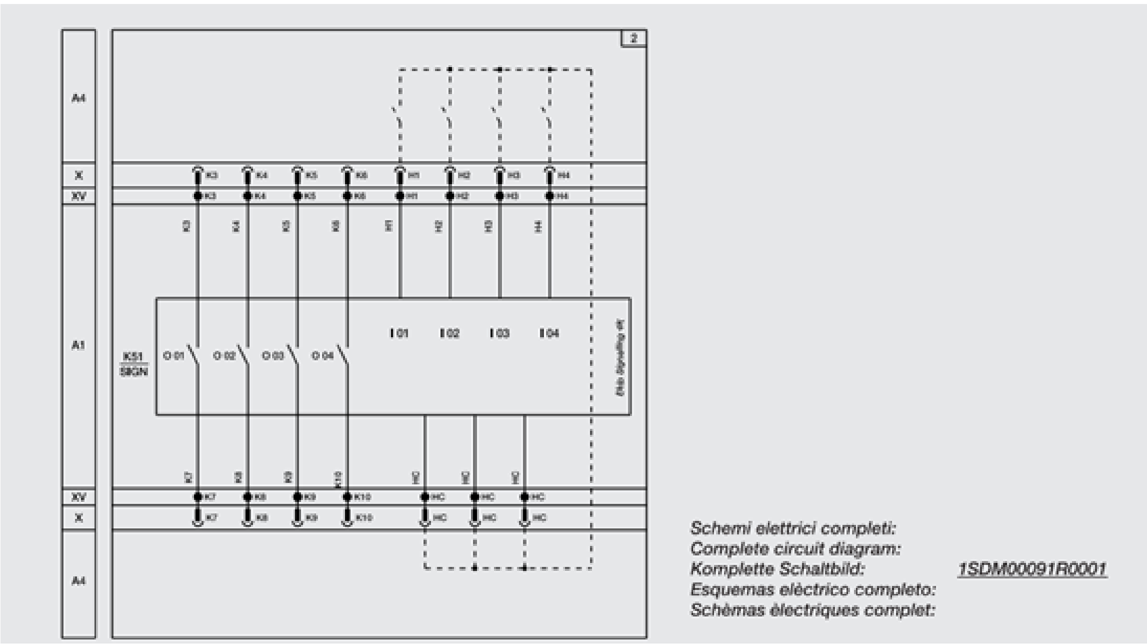

Figure 8.21. Electrical scheme of the 4K module

Figure 8.22. Instrumentation used in the test n.3

- Replace "Ekip Signalling 4K present" with "Ekip Signalling 4K.1 IN" to be turned on by the switch when the dryer electronic board turns on.
- Replace "CB Closed" with "Ekip Signalling 4K.2 IN" to be turned on when tumble dryer is cycling.
- Replace "Ekip Signalling 4K.3 OUT" with "Ekip Signalling 4K.3 IN" so that you manually determine if the frequency is below or above the threshold by closing or opening the switch, respectively.

In Ekip Connect, as already discussed, assign to output 4K.1 Out the power on command if one of "PLC Out B" or "PLC Out C" is high and assign to 4K.2 Out the power on command if "PLC Out A" is high.

The complete instrumentation used for this test is shown in the image 8.22, and once set up as above, the following tests were performed:

1. After power on, if FW1 alarm trips → nothing happens.
2. After I start it manually and activate FW1 alarm for at least 5 seconds → it stops.
3. After it is paused if I deactivate and reactivate FW1 alarm → nothing happens.
4. After it has stopped because the frequency protection has been activated, when I turn off FW1 alarm, if more than 45 seconds but less than 2 minutes have passed → it will resume the cycling.

5. After automatic restart, if FW1 alarm is activated, at least 45 seconds must have elapsed for it to stop automatically again → it stops after 45 seconds.

6. After automatic stop for protection, if the power goes out, when it comes back and there is no more the alarm from FW1, if less than 2 minutes have passed → it resumes.

7. If I manually turn it off or pause it within 2 minutes after the frequency protection command → it will resumes.

8. If there is a power failure while it is running, if the frequency protection was not triggered, when power is restored → nothing happens and the dryer remains in pause state.

9. If the power goes out after the protection has been triggered and the power comes back on within 2 minutes → it resumes the program.

In order to evaluate if it is really worth to make the hardware modification for the on/off sensor of the electronic board of the dryer, let's try to exclude it from the logic - thus keeping the switch associated to 4K.1 IN always closed -:

1. → ditto.

2. → ditto.

3. → ditto.

4. → ditto.

5. → ditto.

6. → it depends on which on/off state it is in when the minimum time of 45 seconds elapses and the FW1 signal is deactivated → if on it will resume/if off a start signal is sent but it has no effect and once power comes back it remains paused.

7. if it is turned off → it does not resume (the signal is sent but the dryer is off). if it is paused → ditto (it resumes).

8. → ditto.

9. → nothing happens (the signal is sent when the power goes out and when the power comes back it remains paused).

Therefore, thanks to these tests, it was established that the probe that senses the on/off status of the dryer can be excluded from future considerations, while the way to establish whether the drying cycle is running or not becomes fundamental. In order to establish this, I chose to measure the current drawn during the various drying phases and for different programs of use. Once the minimum current absorbed by the

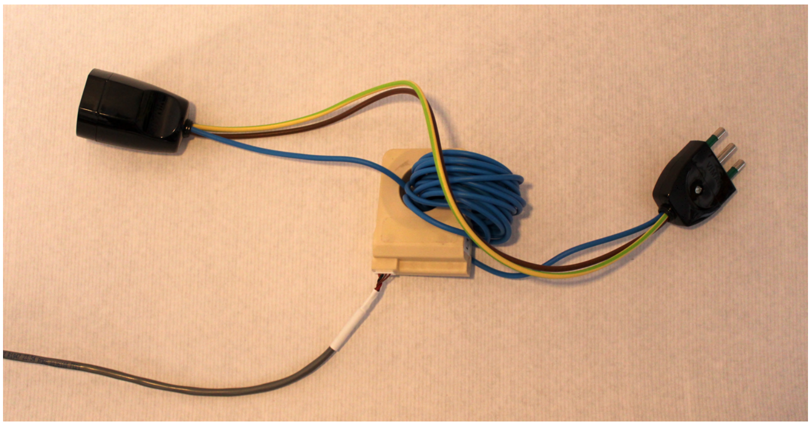

Figure 8.23. Picture of the phase cable wrapped around the current sensor

dryer during the execution of all its cycles has been established, similarly to what has been done for the mains frequency, a current alarm is programmed to be activated when a set threshold is exceeded by a value lower than the lowest current absorption. Thanks to this mode, according to the state of the alarm, the operating status of the load to be controlled is determined.

8.7 Test n.4

Analyzing the electrical current consumption for different drying cycles, a minimum current consumption of about 0.5 A was obtained with refractory periods of maximum 4 seconds between one phase of the drying cycle and the next. As a current probe, a toroidal ring sensor was used, connected directly to Ekip UP Pro. Thus constructively I modified the connection of the electrical outlet of the dryer by adding an extension cord with the phase, neutral and ground cables separated. In this way it was possible to pass the phase cable into the loop of the current sensor for measuring the absorbed current. Since the minimum current during operation reaches values of 0.5 A and the minimum threshold for the current alarm that can be set in the Ekip UP Pro device is 10 A, 23 windings of the phase were made around this ring probe. In this way, the measuring device perceives just under 12 A as the minimum current absorbed by the appliance. These elements are represented in the picture **??**.

In the Custom Logic it has been programmed as shown below, making as modifications a delay on the evaluation of the start/pause state ("Iw1 prealarm") through a DELAY ON of 3 seconds - for the pause command - and of 10 seconds - for the resumption command - to avoid confusing the logic because of particular fluctuations or of the measured refractory periods in which although the absorption is zero, the program is still running since it is simply between one phase and another of a drying cycle. The only thing that is now simulated is the drop in mains frequency below the theoretical threshold using the 4K.1 IN input connected to a switch. The complete instrumentation is shown in Figure 8.24.

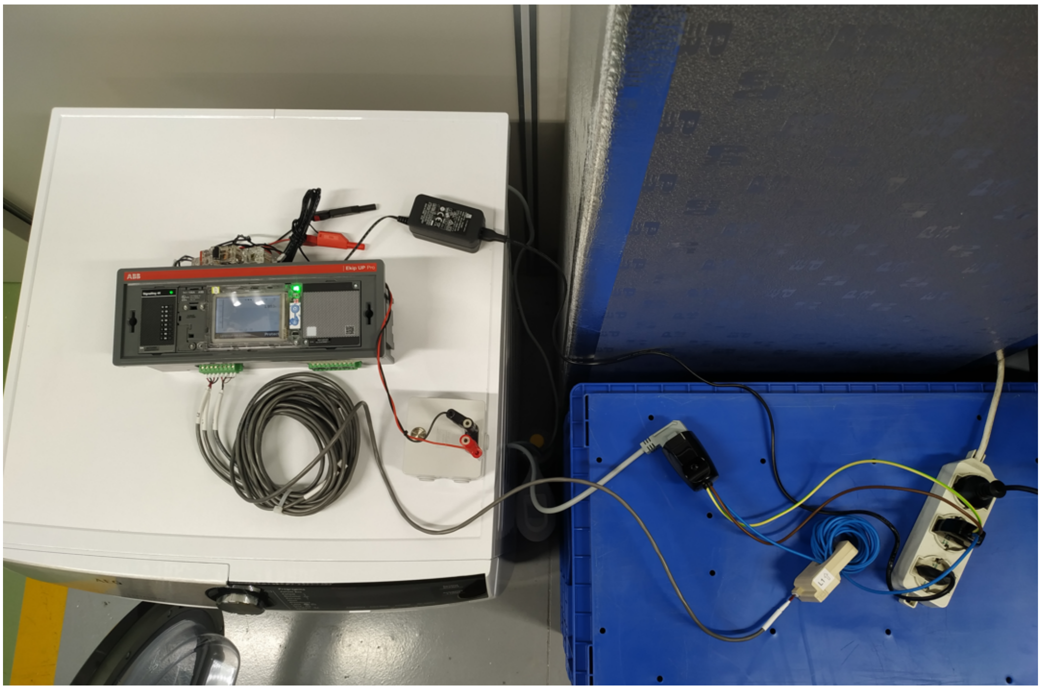

Figure 8.24. Picture of the instrumentation used in the test n.4

Since it was determined to eliminate the on/off state valuation of the device, the logic has lightened and only three rules remain. These rules are schematized in the snapshots 8.25, 8.26 and 8.27.

When assigning parameters:

- To set the alarm current threshold, we need to go to “IW configuration” of the “Other Settings A” section of the “Protection Parameters” menu as shown in Figure 8.28. Since I am using an Ekip UP Pro with a 100 A rated module installed, the minimum treshold of 0.1 the current nominal value corresponds to 10 A.
- Assign the 4K.2 OUT to high status if “PLC Out B” or “PLC Out C” are

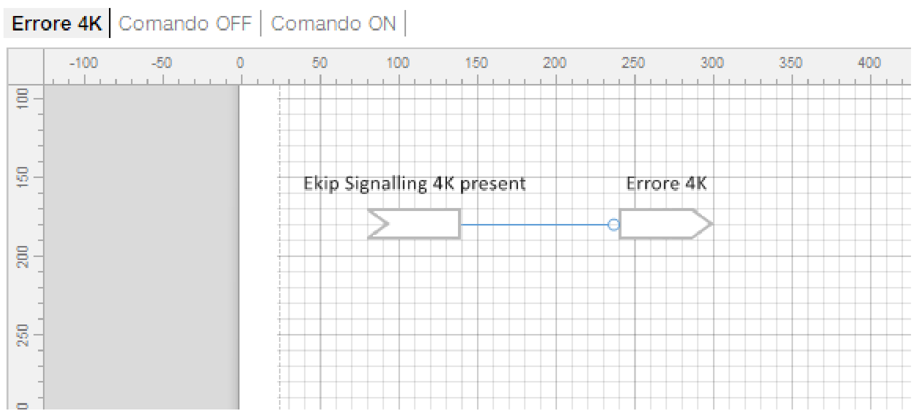

Figure 8.25. Rule in case an error in the 4K module occurs for test n.4

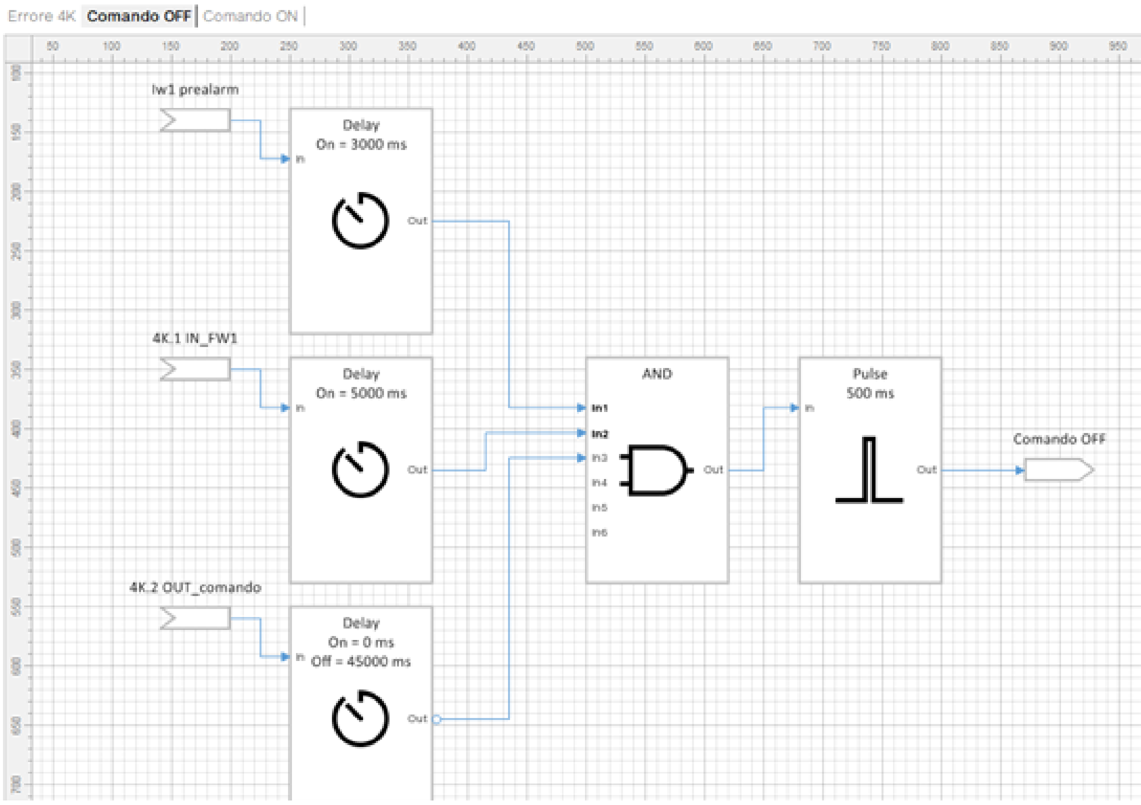

Figure 8.26. Command off rule for test n.4

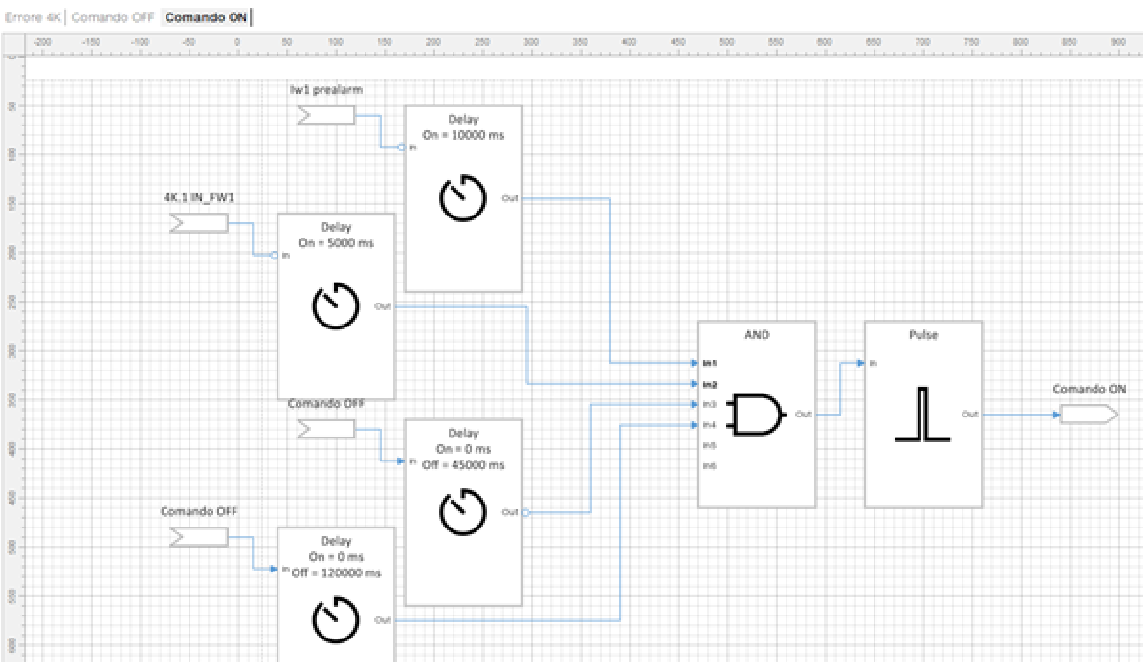

Figure 8.27. Command on rule for test n.4

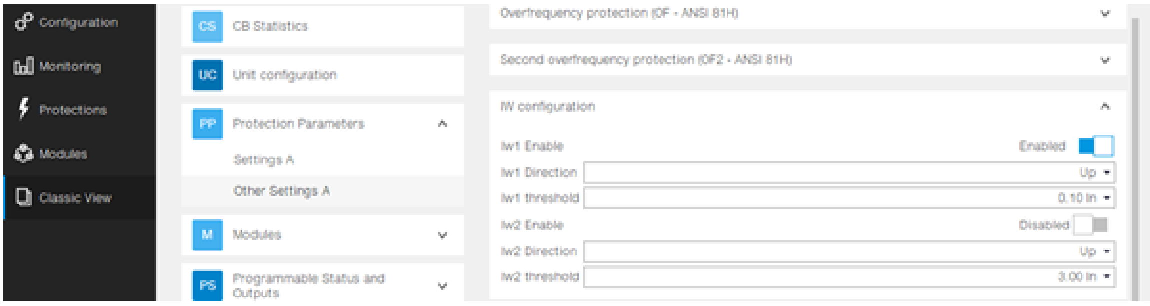

Figure 8.28. Value assignment to the current threshold IW1

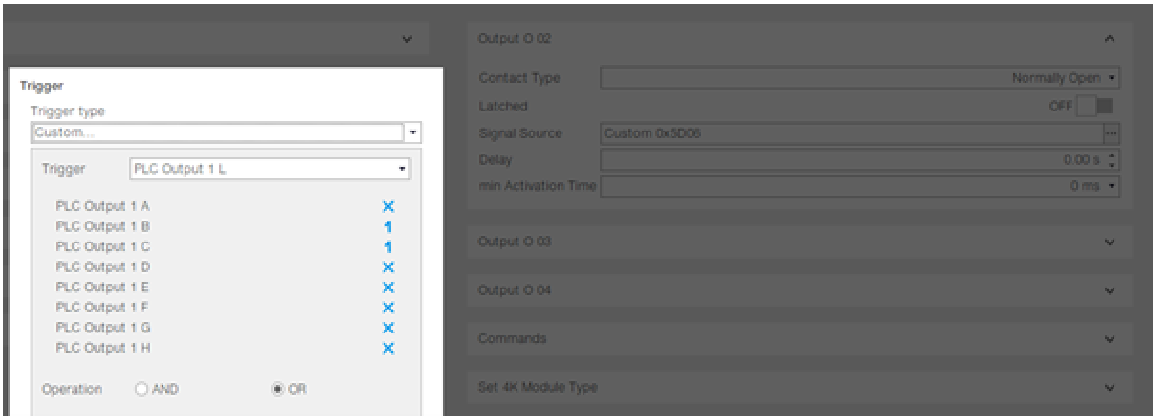

Figure 8.29. Assignment of the output command for test n.4

verified - as shown in the figure 8.29 - and make the physical connection to the custom electronic board mounted on the tumble dryer.

- Assign to 4K.3 OUT the high level status if “PLC Out A” occurs (it signals if there are problems with the 4K module) - as shown in Figure 8.30 -.

The results of this test were satisfactory with regard to the evaluation of the load state, but unsatisfactory in the case of particular scenarios, especially at the end of the work cycle. In the next section we will take a closer look at the load curves and the dryer’s own mode of operation on which the control logic must be adapted.

8.8 Model load absorption curves

In the practical realization of this project, it was not possible to completely analyze the functioning of the dryer taken as model of all programmable loads, because it offers a wide range of drying programs with the possibility of further modifying some of their settings. I preferred, therefore, to concentrate on the programs of greatest practical use of the appliance, imagining what the most used ones might be.

The programs thus considered are five and were tested according to the table 8.3.

For these working cycles, the current absorption was measured minute by minute using a toroidal sensor - described in the previous section - connected to the Ekip UP

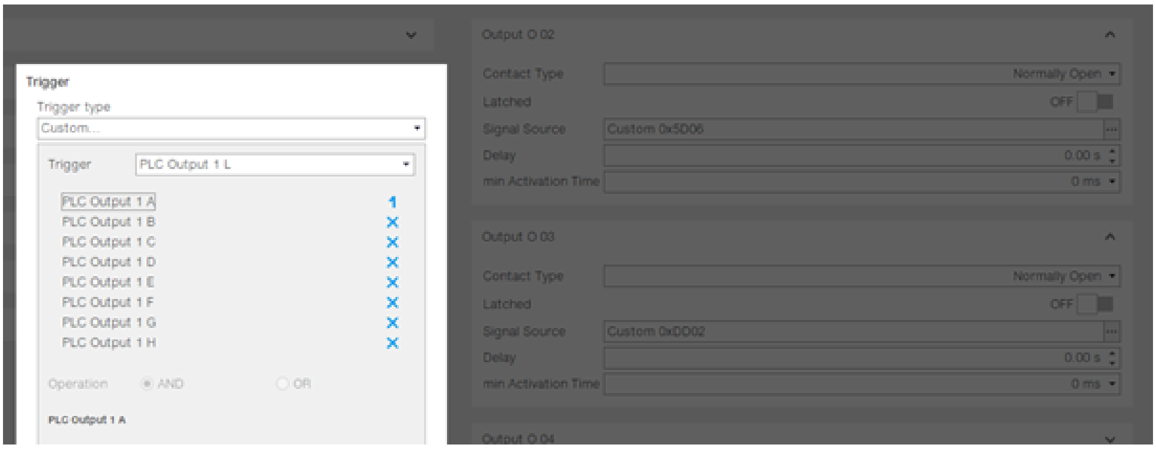

Figure 8.30. Assignment of the 4K module error for test n.4

Table 8.3. Testing modes of the tumble dryer programs

Program	Duration	Load	Water	Current consumption curve
ECO Cottons	02:08	3 beach towels	2 L	Figure 8.31
Cottons	02:10	3 beach towels	1,5 L	Figure 8.32
Extra Silent Cottons	03:02	3 beach towels	1,5 L	Figure 8.33
Mixed	00:55	2 beach towels	1 L	Figure 8.34
Refresh	00:22	1 beach towel	0 L	Figure 8.35

Figure 8.31. Current absorption for the ECO Cottons program

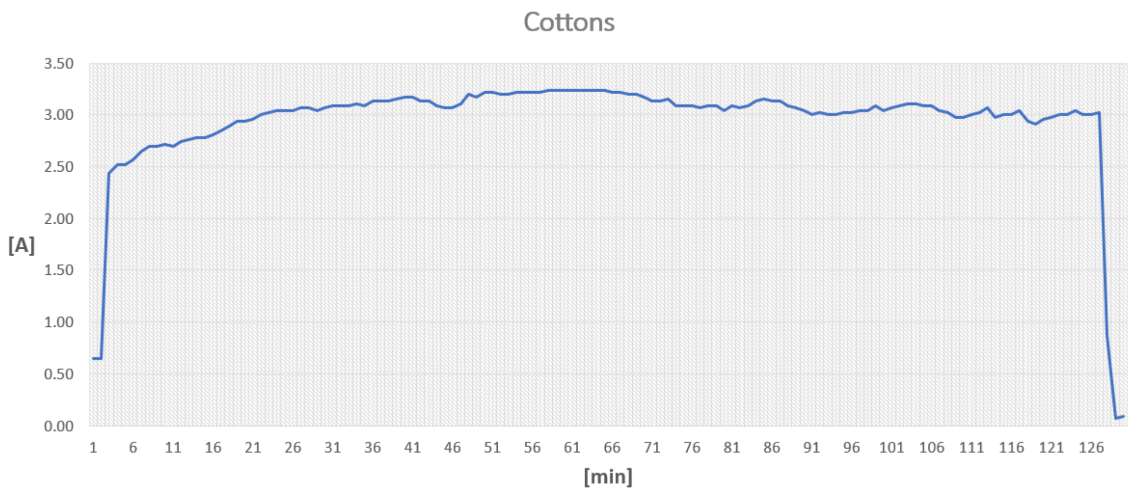

Figure 8.32. Current absorption for the Cottons program

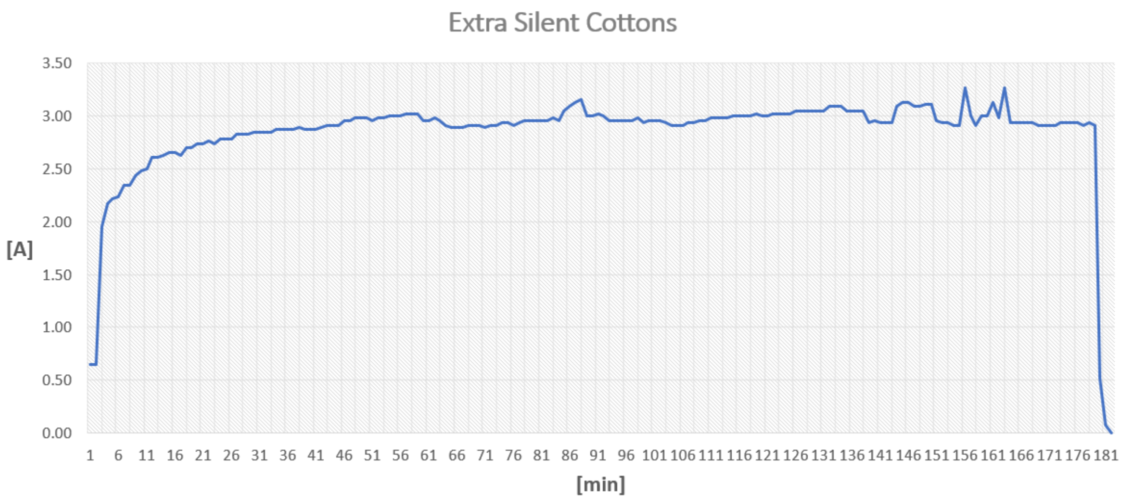

Figure 8.33. Current absorption for the Extra Silent Cottons program

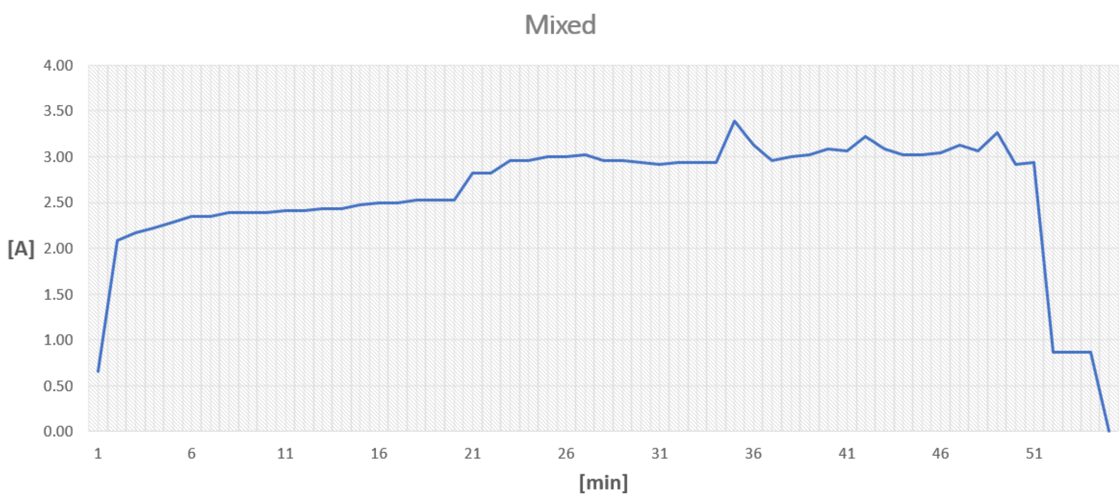

Figure 8.34. Current absorption for the Mixed program

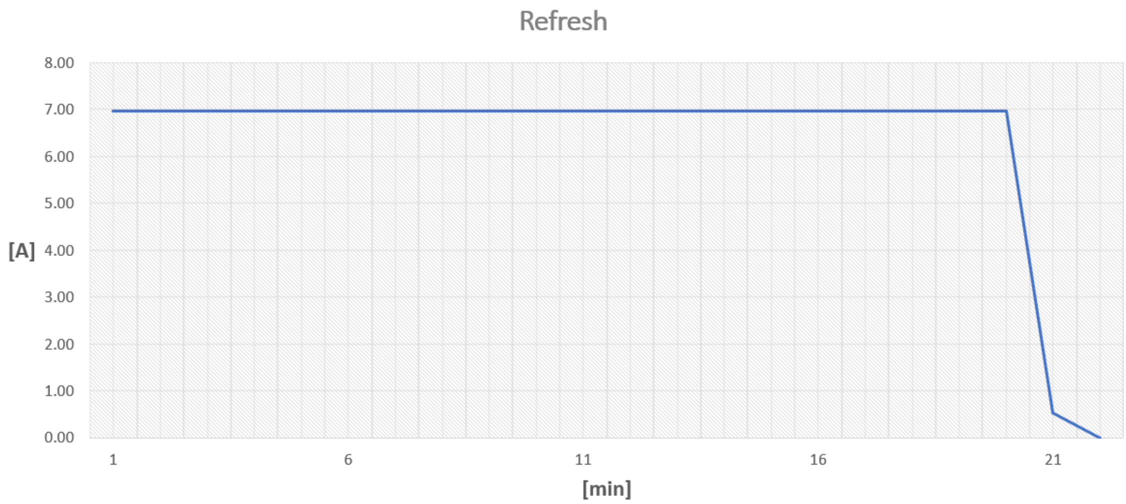

Figure 8.35. Current absorption for the Refresh program

Pro, with the phase cable wrapped 23 times around it. Thus, from the current value read on the monitor of the Ekip UP Pro, to get the real absorption it is then necessary to divide by a factor of 23. For all the work cycles analyzed, during data acquisition, in the minute range the current value read suffered from average oscillations in a range of 2 A with absorption in the order of 60-70 A. Since the uncertainty on the measurement is modest compared to its average value - ascribable to about 3% -, and I, with respect to the real value of the absorbed current, am mostly interested in studying the transients in which the absorption is below the threshold of the current alarm of the Ekip UP Pro, I chose not to take it into account. During these tests, when the dryer is powered on and in a pause condition, a current draw of the actual value of about 0.09 A is recorded. When it then starts up, for all the five work cycles examined, there is a current peak in the first few seconds of operation, varying in value from 0.9 A to 4.5 A. From the instant the programs are started until they are near their end, none of them presents a period of time of more than 4 seconds in which, even though they are running, the current absorption is lower than the set threshold. However, when there are generally two minutes left before the end of the programs, the ventilation cooling system comes into operation. This lasts for a time that is proportional to the internal temperature of the drum, so it may happen that if it is already quite low, for the entire duration of these two minutes, the current absorption is below the set threshold, although the work program is still running. This could fool the control logic into thinking that the dryer is paused when it is not. To be sure that the dryer under examination is actually in pause, therefore, it is necessary to wait at least 2 minutes from when the current drops below the set threshold and at the end of which, if it has not risen again in the meantime, it can be said with certainty that the appliance is in pause. At the end of the ventilation phase, the anti-fold phase comes into operation. This phase has a minimum duration of 30 minutes, with the possibility of increasing its duration by the user, in which

30 seconds of almost zero absorption (0.09 A in reality) alternate with 4 seconds of absorption of about 0.65 A.

During these tests, the behavior of the appliance when the pause/start button is activated was also evaluated. When the dryer is turn on – by pressing the physical power button - and we proceed to choose the program to be used, if we do not manually start it, or do not program a delayed start, after about 5 minutes and 10 seconds the electronic board turns off and we have to press the physical power button again. On the other hand, our control logic can work because once a drying program is successfully started, if we pause it, the electronic board remains on no matter how much time passes - physically I tested this time for approximately 45 minutes, a long enough time to think that the network frequency can return to its nominal value and the control logic will resume the appliance before it shuts down due to inactivity -. At the end of the work cycle, during the last anti-fold phase, if we press the pause button, it will pause as if it had finished its work for good. If we do not press it again for about 5 minutes and 10 seconds, similar to when we select but do not start the program, the dryer will turn off due to inactivity. However, if we happen to press it again before this time has expired, the circuit board will reset itself to start a new work cycle from the beginning. Therefore, if we press the start/pause button once again, it will restart the program previously selected from the beginning. This eventuality is quite remote, because it must occur that just in the anti-fold phase, the load pause command is sent with consequent resumption signal once the frequency returns above the threshold. After that, before the automatic shutdown time has passed, it must happen that there is a new frequency drop, with consequent rise, so that a new command is sent to restart the working cycle from the beginning. To avoid this unpleasant hypothesis, it is necessary to wait at least 5 minutes and 20 seconds between the pause command and the resumption one. So, if the tumble dryer is at this stage of the program, the electronic board will shut down for inactivity before it can receive any resumption command.

8.9 Final Custom Logic for tumble dryer T8DEC94ST 8000 Series 9 Kg model by AEG

Following all the tests carried out and the characteristic behavior of the load on which this frequency control logic is being installed, I have arrived at the final form that the Custom Logic must have for proper operation. It is necessary in case of connection problems of the 4K module predisposed to the communication between the logic and the load to be controlled, to communicate an error message to the user - Figure 8.36 -.

The pause command - shown in the figure 8.37 - should be sent as quickly as possible after it becomes apparent that there is a problem with the network frequency. In order to establish that the load is switched on and in operation, it is sufficient that

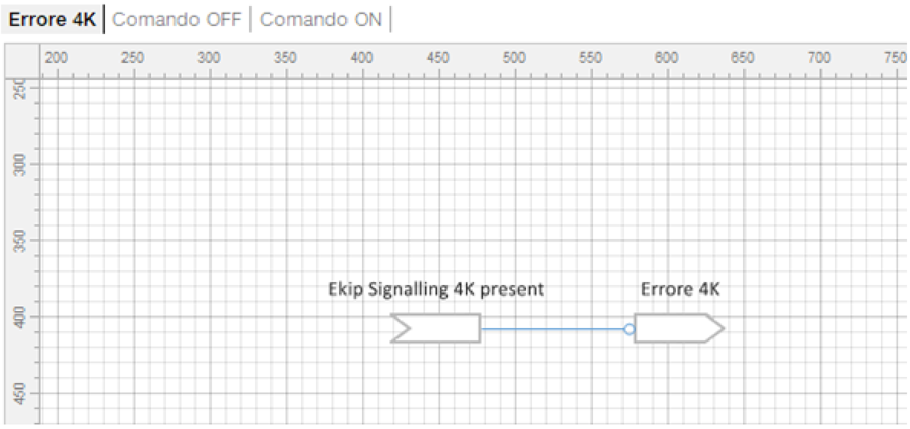

Figure 8.36. Final rule in case an error in the 4K module occurs

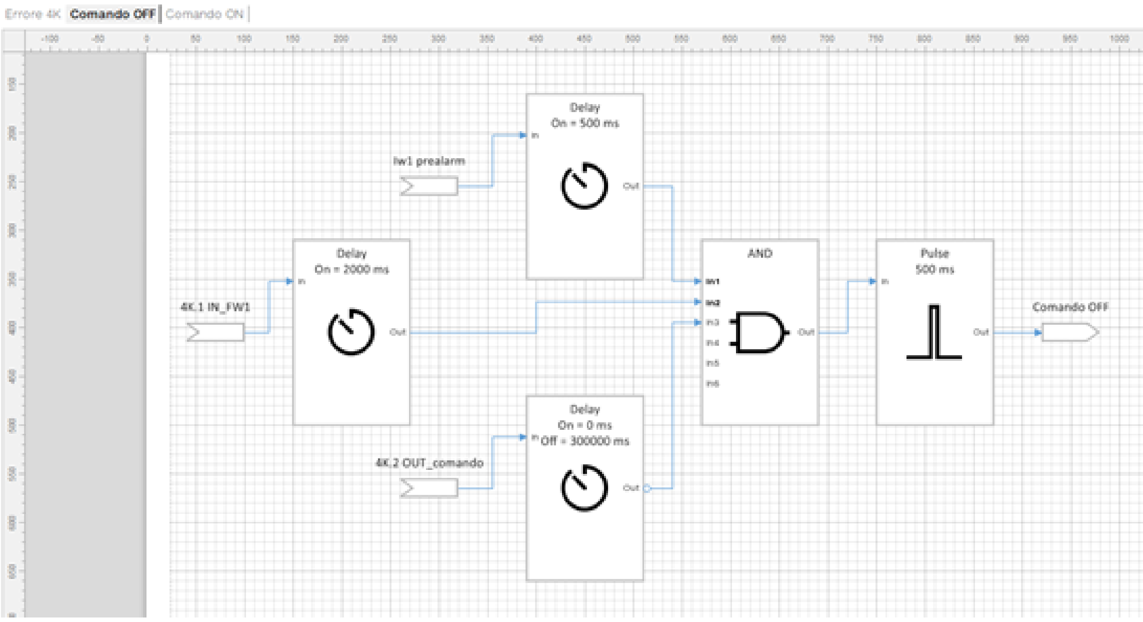

Figure 8.37. Final command off rule

the measured current is above the set threshold for at least 500 ms ("Iw1 prealarm"). A problem on the electrical network is recognized when the frequency alarm remains active for at least 2 seconds ("4K.1 IN_FW1"). Finally, in order to preserve the good work of the dryer, at least 5 minutes of uninterrupted operation must have passed before a pause command is sent ("4K.2 OUT_command").

The resumption command - Figure 8.38 - must be sent only when the load is not running. To recognize it, as stated in the previous section, the current must be below the minimum threshold for at least 2 minutes ("Iw1 prealarm"). It must be sent with the certainty that the frequency has returned close to its nominal value, i.e. allowing a minimum time of 60 seconds to elapse since the frequency alarm is deactivated ("4K.1 IN_FW1"). To prevent an unintentional restart of the selected program, a minimum time of 5 minutes and 20 seconds must have passed since the

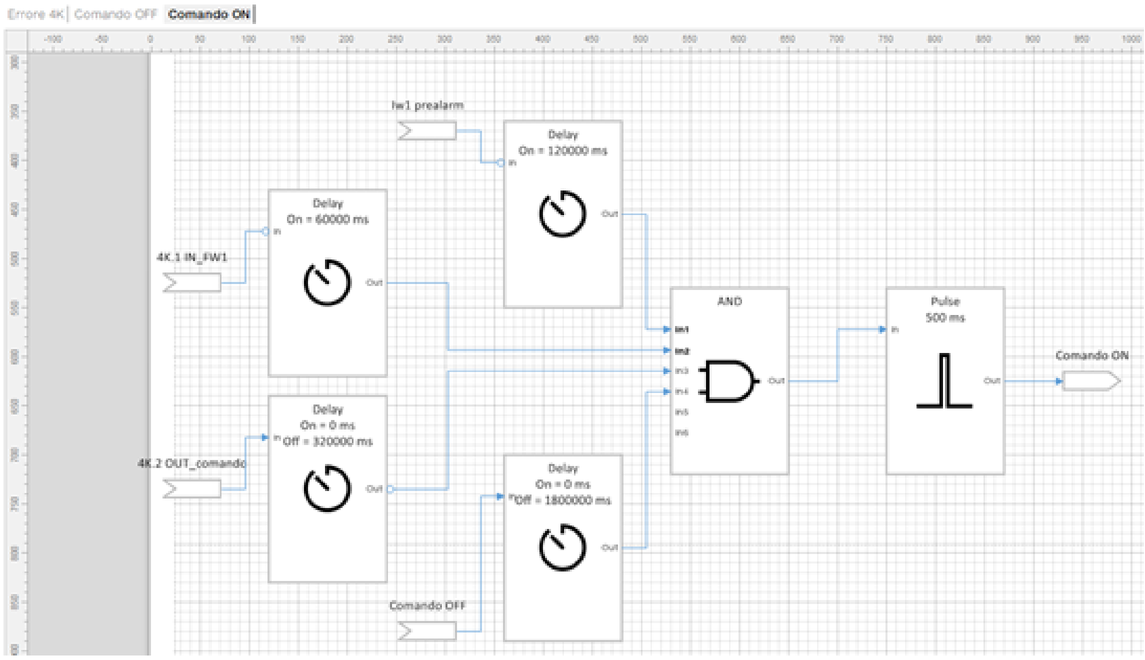

Figure 8.38. Final command on rule

previous pause command ("4K.2 OUT_command"). Obviously, it can be activated only after the pause command has come into operation, within a time window in which the network frequency has certainly returned above the alarm threshold and which has been estimated to be 30 minutes ("Command OFF").

Performing some test runs, this time the behavior is completely satisfactory and reflects the desired operation without any kind of problem. Therefore, for the loads that have been defined as programmable, we can consider this control logic for primary frequency regulation successfully concluded and potentially applicable for the future.

Chapter 9

Custom logic for thermal loads

9.1 General premises

Contrary to what happens with the tumble dryer and with all programmable loads, as already mentioned in Section 7.3, the command coming from the output of the 4K module does not act directly on the pause or start status of the appliance, but acts on the temperature selection set by the user. In case of problems in the electrical network, with consequent lowering of the frequency below the nominal value, in the certainty that the generic thermal load stops absorbing electrical energy, it is not sufficient to lower the temperature by one step - in the specific case of the oven under examination equivalent to a variation of 20° -, but to lower it by many steps until a lower temperature is selected with respect to the one the appliance has at that particular moment. Statistically speaking, it must be said that both for our oven and for all other thermal appliances, when they are turned on, most of the operating time is spent in thermal steady-state, therefore with small variations in temperature with respect to the set temperature. For this reason, it can be assumed that by varying the temperature set by the user even by not much, the effect of turning the appliance on or off is obtained - i.e. it is possible to control its current absorption in case of need -.

For the thermal loads too, I have chosen to give a continuity of reasoning as done for the programmable loads, that is: starting from the programming of a logic that, taking into account the premises previously stated, can be the basic structure for the generic thermal appliance, and then adapting it only later to the specific case of the model taken into consideration - i.e. the oven model FA3S 844 IX HA by Hotpoint/Ariston -. In the case of thermal appliances, this time I can decide not only to force the switching off, but also to force the switching on. This, by virtue of the fact that, unlike programmable appliances, they are perpetually on and working. Therefore, considering a heating load, such as an oven or a water heater, there will be four commands of different nature:

- the command that lowers the temperature when the network frequency is lowered to force its shutdown;

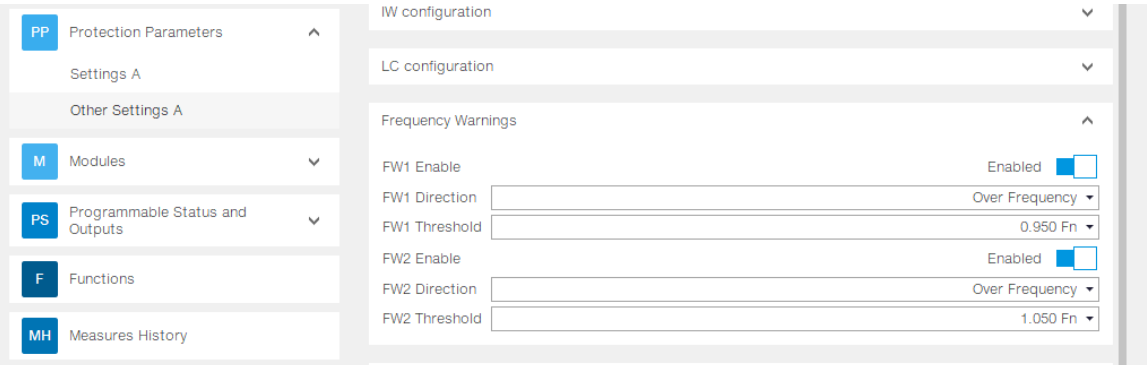

Figure 9.1. Set of the over-frequency alarm

- the command to reset the temperature after the frequency has returned above the threshold;
- the command that raises the temperature in case of increase of the network frequency to force its switching on;
- temperature reset command after the frequency has returned below the threshold.

If the control logic is instead to be adapted to a refrigerating thermal load, the temperature variation commands will be inverted; i.e. to turn the appliance off it will be necessary to raise the temperature, to turn it on it will be necessary to lower it.

9.2 Custom Logic programming for generic thermal load

For the programming of this logic it is necessary to set the Ekip UP Pro in the same way as done in the Section 8.1 of the chapter dedicated to the control of programmable loads. Considering that with this category of loads it is also possible to manage cases of over-frequency, first of all it will be useful to establish a new threshold. As done for the under-frequency one, in "Other Settings A" of "Protection Parameters" menu, set the over frequency alarm threshold FW2 to 1.05 Fn - Figure 9.1. Since the nominal frequency is 50 Hz, this new threshold value is 52.5 Hz.

Once the changes have been made and all the necessary steps have been taken into account, it is possible to start the programming through the Custom Logic. To program the command in order to decrease the temperature - and therefore force the shutdown of the load - if the frequency falls below the threshold, in theory it is not really necessary to know whether the load is running or not - i.e. whether it is or it is not absorbing current at that precise moment -. In fact, if the load is off, even though a lowering of the set temperature would not cause any change, it could still be an advantage because it would delay the possible future switch on in case the frequency has not yet had a chance to recover around the nominal value. The

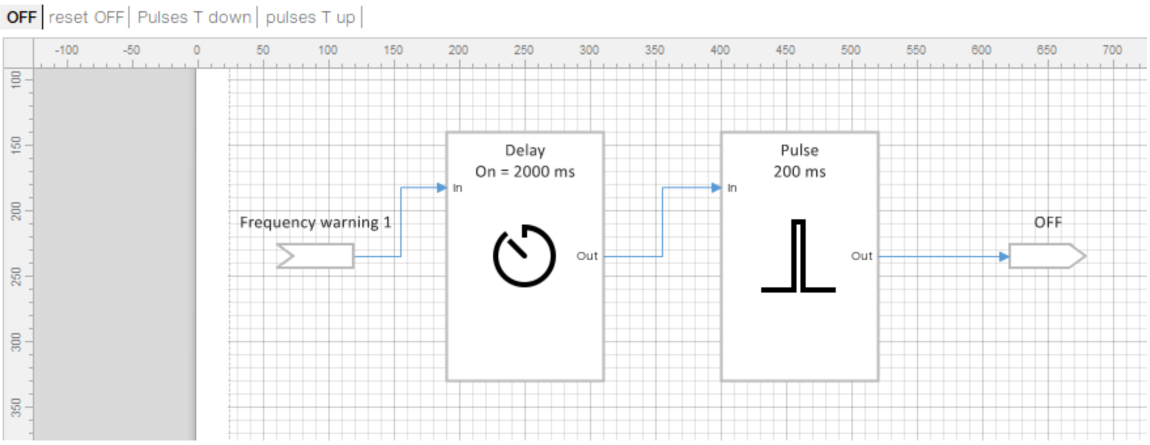

Figure 9.2. Command off for a generic thermal load in case of under-frequency

consequences of this choice also bring benefits with regard to the study of loads and software modifications of the device being designed. For the thermal loads, in fact, it is no longer necessary to know their absorption curves since no threshold value of the absorbed current has to be set - for the counterpart of the programmable loads it was a necessary step instead -. Therefore, for this command - Figure 9.2 - it is sufficient to send a signal that will be associated with the lowering of the temperature ("OFF") by means of a PULSE block with an arbitrary value of 200 ms, after the network frequency has been below the set threshold value for at least 2 seconds ("Frequency warning 1").

This logic involves sending a signal that will be related to the generation of a single notch decrease on the temperature value. A priori, for the generic load, it is not possible to know either the absolute value of this variation, or whether this variation is sufficient to force a switch-off of the appliance. Therefore, it is necessary to program a logic that, after the first command that decrease the set temperature, sends other equal commands, of a number that guarantees both the shutdown and the maintenance of performance, and that can be customized during the installation phase. In order to obtain this function, every time a temperature lowering command is sent ("Ekip Signalling 4K.2 OUT"), signals are generated through the combination of PULSE blocks of 250 ms and DELAY blocks of 200 ms, which alternately activate and deactivate an AND port that has as output the sending of new temperature lowering commands ("Pulses T down"). For the times set in PULSE and DELAY blocks, a new command is sent every 400 ms. To control the number of commands to be sent, at this point it is sufficient to activate the AND port only for a defined period of time starting from the first command sent ("OFF"). Supposing, for example, that we want to send commands that lower the temperature by 10 notches, following the "OFF" command we will put a PULSE block of 4000 ms as shown in the figure 9.5.

Once the temperature lowering commands have been sent, after the frequency has returned above the threshold value, it will be necessary to send the same number of commands but in such a way as to re-establish the initial temperature value. To do

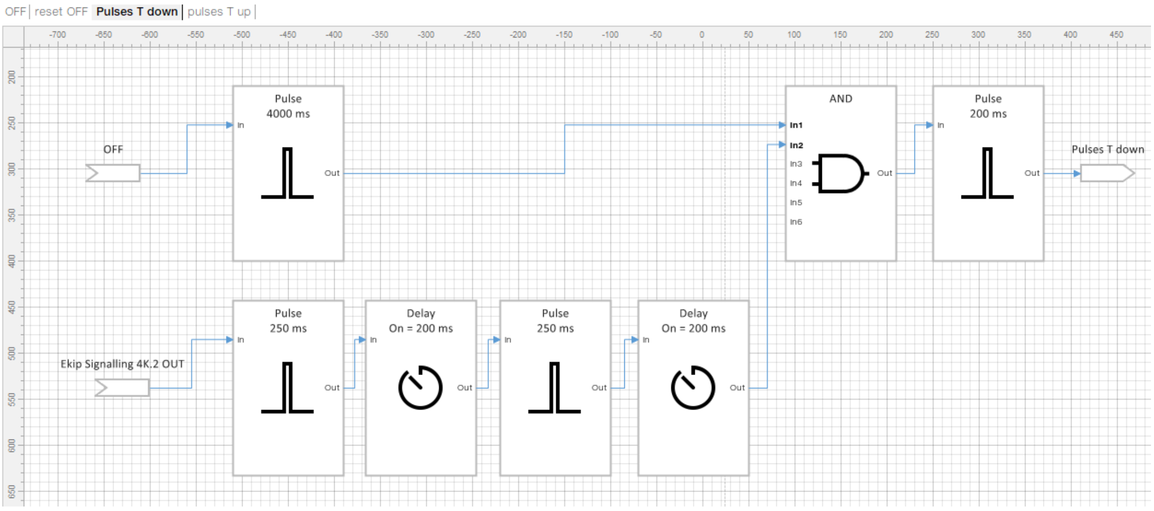

Figure 9.3. Repetition of the command off to decrease the temperature more in case of under-frequency

this, the first temperature reset command ("reset OFF") will have to be sent only after the "OFF" command has come into operation - similarly to what has been done for the tumble dryer resume command - and after the mains frequency alarm ("Frequency warning 1") has been deactivated for a minimum time of 1 minute - Figure 9.4.

Following the first command for raising the temperature, subsequent commands of the same number as the commands previously sent for lowering the temperature must be sent. As can be seen from Figure **??**, the programming for sending these commands is entirely analogous to that previously developed in Figure 9.5.

In order to guarantee correct operation, it is necessary that all the time values set in the "Pulses T up" rule must be equal to those set in the "Pulses T down" rule.

For the control in case of an over-frequency phenomenon in the network, the programming is similar to the one done for the under-frequency case but with the caution to invert the assignment of the commands and to use the over-frequency alarm ("Frequency warning 2") as discriminating for the implementation of the temperature control as shown in the figures 9.6 and 9.7.

During programming, however, when more than one command must be sent to trigger more than one temperature notch, assuming that the same temperature variation used for the under-frequency case is to be applied, it will be sufficient to put an OR port in the open time window condition that also takes into account the over-frequency commands ("ON" and "reset ON") - Figure 9.8 and Figure 9.9 -.

9.3 Final Custom Logic for oven model FA3S 844 IX HA by Hotpoint/Ariston

The Custom Logic described in the previous section is applicable to any thermal load for which it is possible to vary the temperature set by the user through the

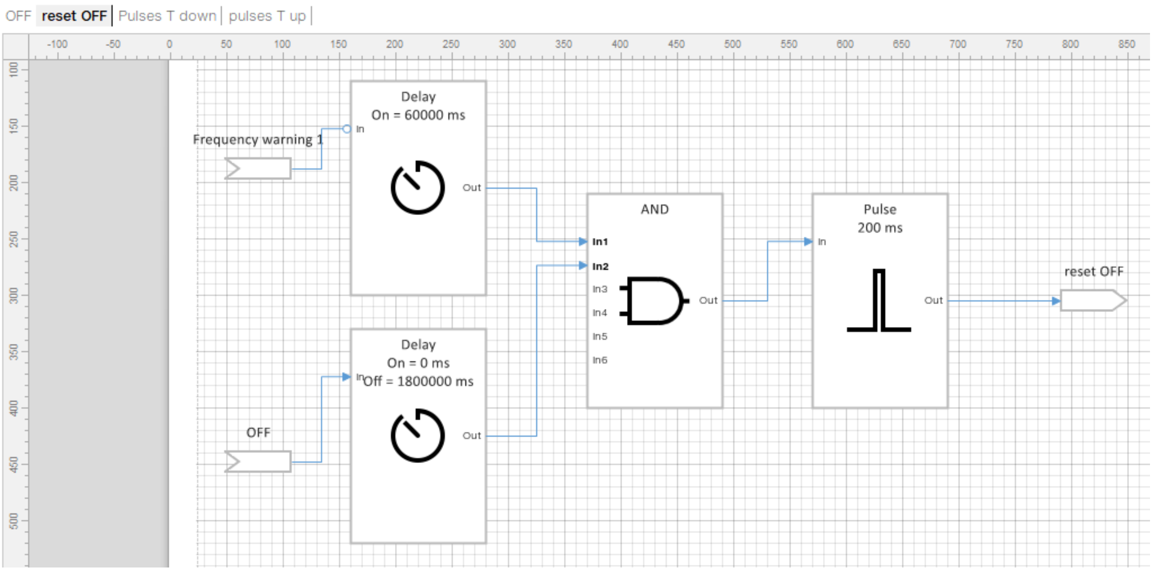

Figure 9.4. Command to reset the temperature for a generic thermal load in case of under-frequency

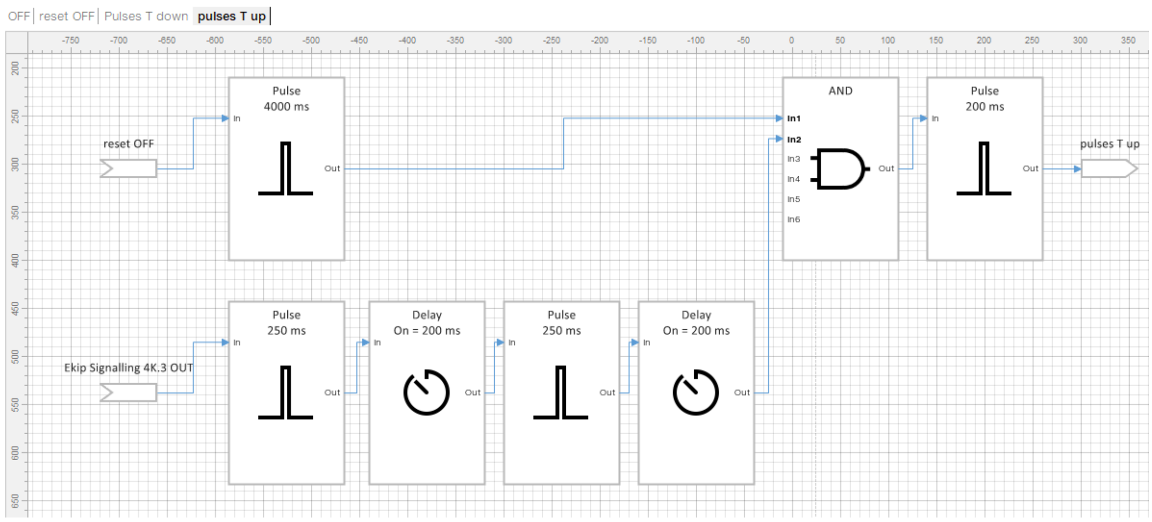

Figure 9.5. Repetitions of the reset command to increase the temperature more in case of under-frequency

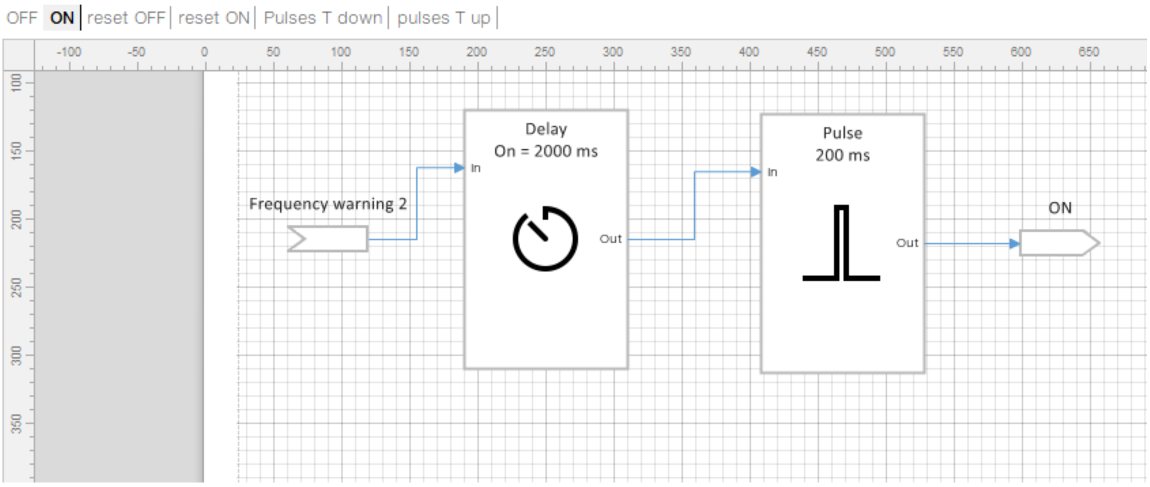

Figure 9.6. Command on for a generic thermal load in case of over-frequency

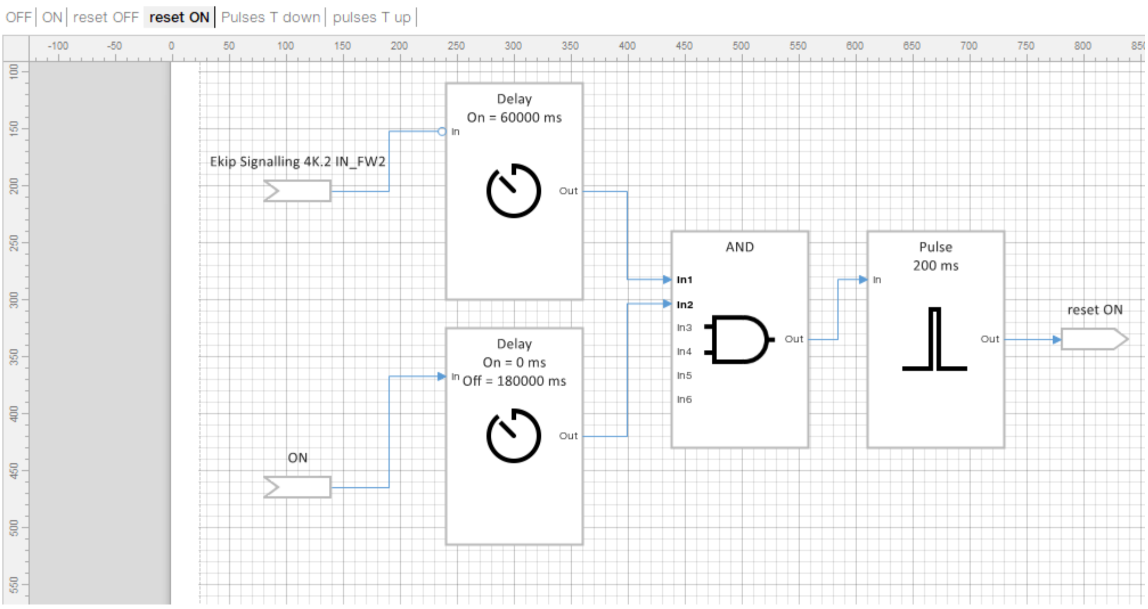

Figure 9.7. Command to reset the temperature for a generic thermal load in case of over-frequency

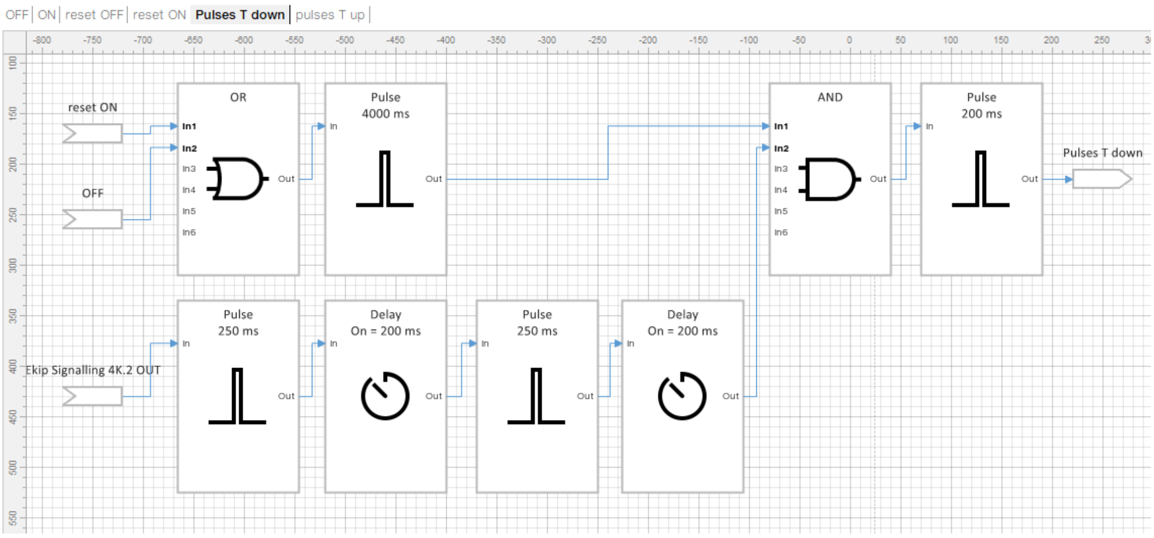

Figure 9.8. Repetitions of the command to decrease the temperature more in both cases of under and over-frequency

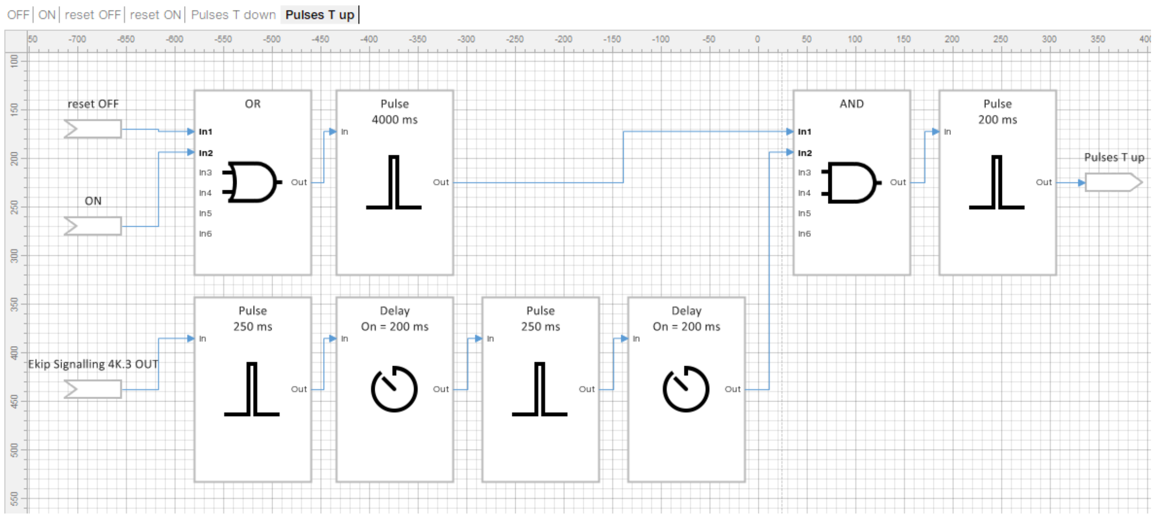

Figure 9.9. Repetitions of the command to increase the temperature more in both cases of under and over-frequency

Figure 9.10. Picture of the instrumentation used to test the oven control logic

contacts of the 4K module of the Ekip UP Pro. It is only necessary to know for a single command to which temperature variation value it corresponds, so as to evaluate the total number of closures of the output contacts of the 4K module. In the case of our oven, one command corresponds to a change of 20°. In the evaluation phase, therefore, a single command is sufficient to guarantee a switch-off or a switch-on of the appliance. In the logic already implemented it will be sufficient to modify in the rules "Pulses T down" and "Pulses T up" the time of the PULSE block that activates the AND gate. Setting a time equal to 400 ms will guarantee the sending of only one command, while keeping the rest of the logic unchanged. Once this change was made and the Custom Logic was uploaded on the Ekip UP Pro, some tests were carried out to analyze the operation of the control system - Figure 9.10.

Simulating through two switches the activation of the two alarms of over-frequency and under-frequency - in a similar way to what was done with the dryer -, the behavior of the oven was then evaluated. The results obtained from these tests were completely satisfactory, taking into account the initial assumptions with which I approached this type of control. It has therefore been demonstrated how this control can also be applied to thermal loads.

To conclude, as can be seen from the comparison with what was done for programmable loads, the general control for thermal loads is more homogeneous, with fewer parameters to be modified according to the specific appliance to be controlled because without the need to measure the current it absorbs or the shutdown times for inactivity.

Chapter 10

Programmable load management tool

10.1 Purpose of the management software

The system that was implemented in Chapter 8 serves as stated in the introduction, to strengthen the primary frequency regulation. Specifically, it acts immediately whenever there is an under-frequency problem. Obviously, this type of problem is more likely to occur when there are peaks in the power required by the loads of the electrical network. To make this type of event less likely, we can base ourselves on the fact that nowadays most programmable appliances have a delayed start function, as does the tumble dryer I use as their model. Thanks to this function, in fact, we can decide to make the load work at a desired time in a completely automatic way. Therefore, we can think of implementing a tool that helps the end-user to start the appliance when it is most convenient for him and for the network operators. Thanks to this new management logic, it is possible to suggest to the user to start the work program when there are not predictable peaks of demand from other loads - such as during the night - avoiding overloads to the domestic users and to the network, thus flattening the daily curve of the power demand seen by the generation plants. A tool of this type is also applicable in the case in which the control device for primary frequency regulation has not been installed in the domestic appliance. In fact, it was thought that this could simply be a supplementary support tool. Each home appliance manufacturer could potentially promote the installation of a smartphone application in which a similar programmable load management program is loaded in relation to their products.

10.2 Tool programming

For the development of this software I chose to use the Phyton programming language, as it is one of the most common high-level languages in programming of this type. In its

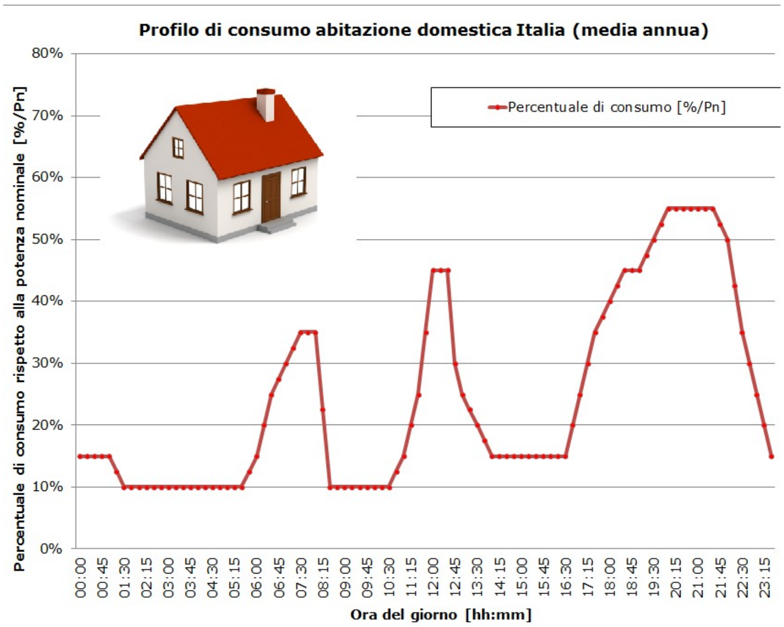

Figure 10.1. Average annual domestic consumption curve in Italy

writing, PyCharm was used as an integrated development environment. The program developed is in principle applicable to all programmable loads, but I have adapted it specifically for the management of my model home appliance described in the previous chapters for the development of the Custom Logic - the model T8DEC94ST 8000 Series 9 Kg By AEG dryer -. The application I want to implement must tell the end user how long to delay the scheduled start of the dryer, after being given as input the chosen work cycle and the time at which he/she wants it to be finished. Based on the expected electrical network status, the program will choose at what time it is most convenient to start the dryer. In the program then, must be defined both the expected absorption curves of the home network of the same day and of the next day, and the absorption curves of the dryer programs. Considering the duration times of the working cycles of the dryer, the time interval that was chosen to be used for discretization, is 15 minutes. Having done this, the average consumption of electrical power absorbed for each quarter hour of operation was calculated. On the other hand, for the expected domestic load curves, which is shown in the figure 10.1, since this is not a study on the calculation of this prediction, I have simply based myself on the Italian representative average of a household of four members in a house equipped with a meter with nominal power 3 kW. [21] The cost of electricity was excluded from the evaluation, as it is too variable, considering the numerous types of contracts that can be signed with the electricity network companies. Once the management program has acquired all the data, after determining the time window in which the work cycle can be started, by means of two nested for cycles, the overall consumption of the mains and of the tumble dryer are compared for each quarter of an hour in which it can be started. Finally, the time window that offers the lowest overall consumption is

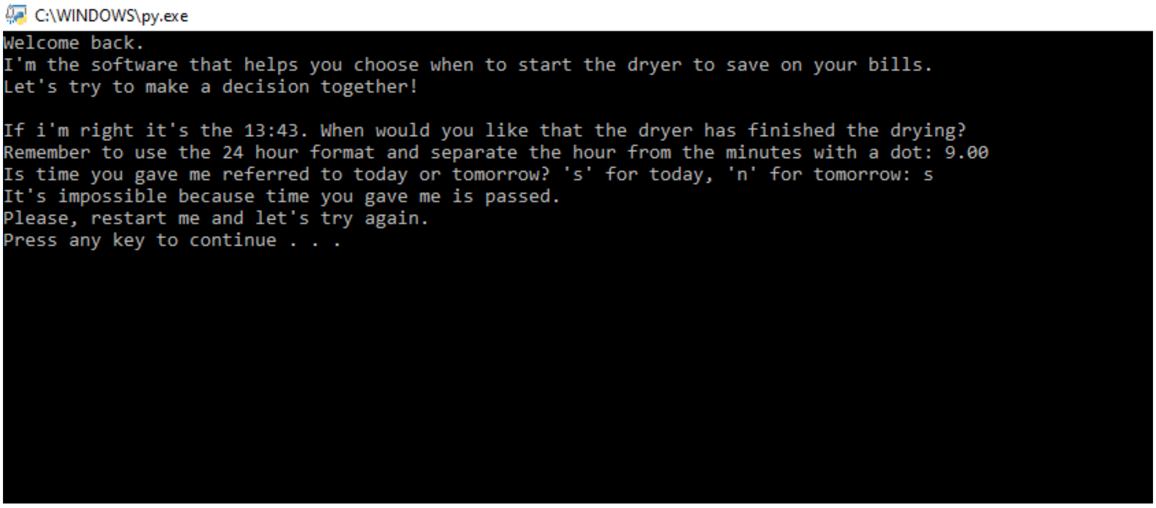

Figure 10.2. Example n.1 of a communication between software and user

selected, the start time is saved and, in relation to the current time, to the user is told how long to delay the programmed start. In this chapter are shown three examples of communication between the machine and the user in relation to possible inputs that the latter may have typed. In the first case - Figure 10.2 -, he typed in a past time. In the second - Figure 10.3 -, he has selected a work cycle that is too long for the set end time. In the last case - Figure 10.4 -, after making several possible typing errors, he managed to get a suggestion as to when to have the dryer start. Finally, for completeness, the entire code that makes up the management program is also shown through images 10.5, 10.6, 10.7, 10.8, 10.9, 10.10, 10.11, 10.12, 10.13.

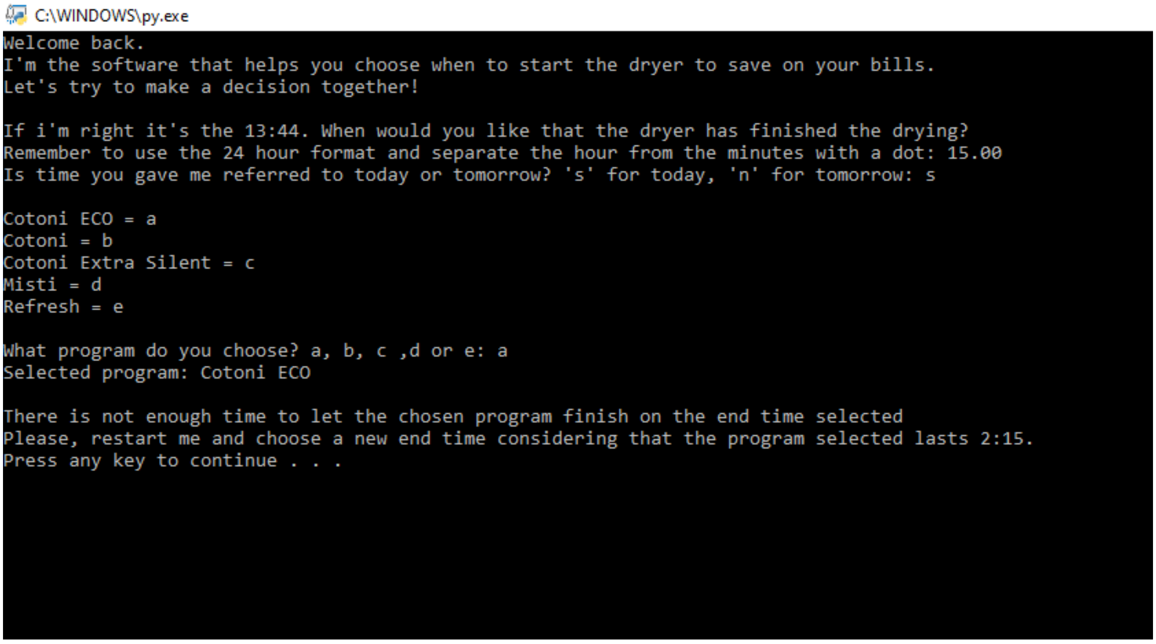

Figure 10.3. Example n.2 of a communication between software and user

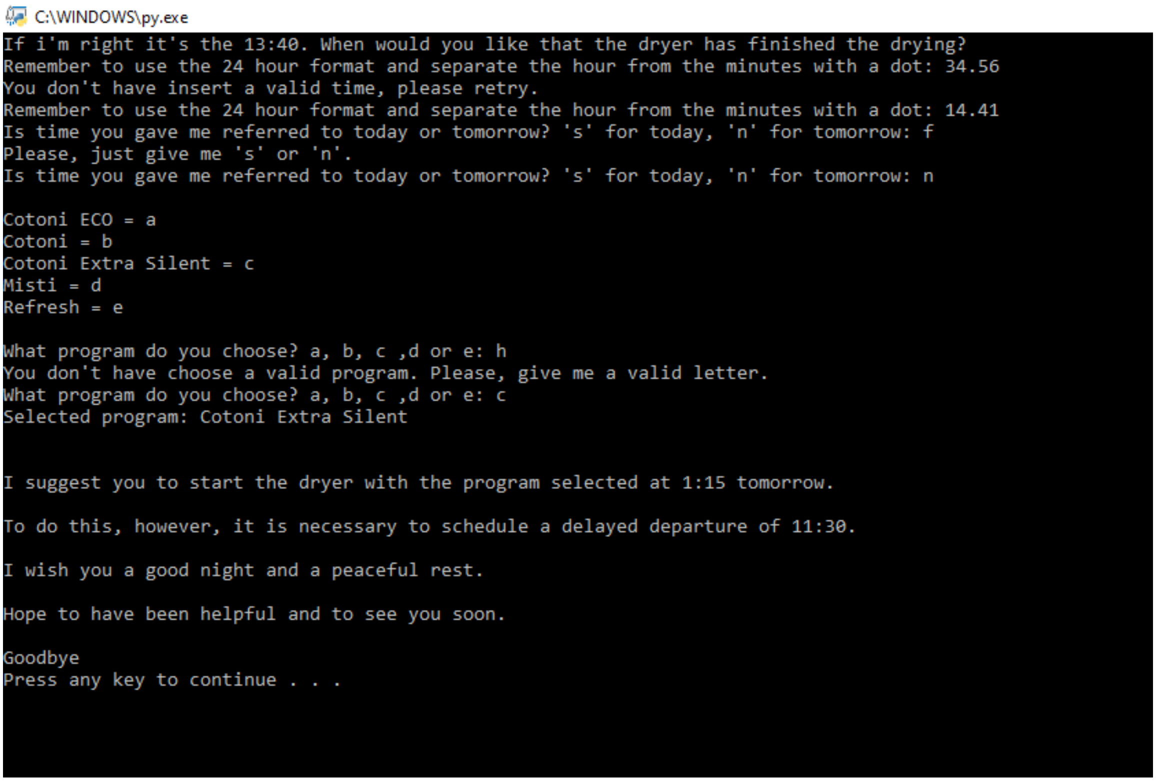

Figure 10.4. Example n.3 of a communication between software and user

```python
import sys
import time
import os
from math import *

#welcome messages
print("Welcome back.")
print ("I'm the software that helps you choose when to start the dryer to save on your bills.")
print ("Let's try to make a decision together!")
print (" ")

#expected consumption for a house with power expressed in kilowatts
#time     = [00:00,00:15,00:30,00:45,01:00,01:15,01:30,01:45,02:00,02:15,02:30,02:45,03:00,03:15,03:30,03:45,04:00,04:15,04:30,04:45,05:00,05:15,05:30,05:
rete_oggi = [0.45 , 0.45, 0.45, 0.45, 0.37, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.37, 0.

rete_domani = [0.45 , 0.45, 0.45, 0.45, 0.37, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.30, 0.37,

#time acquisition and conversion into decimal for the calculation
h_now = (time.strftime("%H"))
m_now = (time.strftime("%M"))

if int(m_now) <= 15:
    m_now = float(0.25)
elif int(m_now) <= 30:
    m_now = float(0.5)
elif int(m_now) <= 45:
    m_now = float(0.75)
else:
    m_now = float(1)
ora_attuale = float(h_now) + float(m_now)
ora = float(ora_attuale)/float(0.25)
```

Figure 10.5. Python program lines 1 - 31

```python

#acquisition of the instant in which you want the program to be finished and conversion into decimal
print ("If i'm right it's the " + time.strftime("%H:%M") + ". When would you like that the dryer has finished the drying?")
while True:
    end = float(input("Remember to use the 24 hour format and separate the hour from the minutes with a dot: "))

    if float(end) < 24:
        break
    else:
        print("You don't have insert a valid time, please retry.")

h_end = int(end)
m_end = float(float(end) - int(h_end)) * 100
if int(m_end) < 15:
    m_end = float(0)
elif int(m_end) < 30:
    m_end = float(0.25)
elif int(m_end) < 45:
    m_end = float(0.50)
else:
    m_end = float(0.75)
ora_fine = float(h_end) + float(m_end)
fine = float(ora_fine)/float(0.25)

#day acquisition
while True:
    in_giornata = (input("Is time you gave me referred to today or tomorrow? 's' for today, 'n' for tomorrow: "))
    if in_giornata == "s" or in_giornata == "n":
        break
    else:
        print("Please, just give me 's' or 'n'.")
```

Figure 10.6. Python program lines 32 - 63

```python

#if the time given has passed
if fine < ora and in_giornata == "s":
    print ("It's impossible because time you gave me is passed.")
    print ("Please, restart me and let's try again.")
    os.system("pause")
    sys.exit()

#dryer program list
print (" ")
print ("Cotoni ECO = a")
print ("Cotoni = b")
print ("Cotoni Extra Silent = c")
print ("Misti = d")
print ("Refresh = e")
print (" ")

#dryer programs power consuption
prog_a = [0.56, 0.65, 0.72, 0.75, 0.78, 0.75, 0.75, 0.75, 0.41]
durata_a = len(prog_a)

prog_b = [0.55, 0.69, 0.72, 0.74, 0.73, 0.71, 0.70, 0.69, 0.51]
durata_b = len(prog_b)

prog_c = [0.50, 0.63, 0.66, 0.69, 0.67, 0.69, 0.68, 0.68, 0.70, 0.70, 0.69, 0.64]
durata_c = len(prog_c)

prog_d = [0.51, 0.64, 0.70, 0.48]
durata_d = len(prog_d)

prog_e = [1.60, 1.16,]
durata_e = len(prog_e)

```

Figure 10.7. Python program lines 64 - 96

```python
#dryer program acquisition
prog = []

while True:
    ciclo = (input ("What program do you choose? a, b, c ,d or e: "))

    if ciclo == "a":
        prog = prog_a.copy()
        print("Selected program: Cotoni ECO")
        break
    elif ciclo == "b":
        prog = prog_b.copy()
        print("Selected program: Cotoni")
        break
    elif ciclo == "c":
        prog = prog_c.copy()
        print("Selected program: Cotoni Extra Silent")
        break
    elif ciclo == "d":
        prog = prog_d.copy()
        print("Selected program: Misti")
        break
    elif ciclo == "e":
        prog = prog_e.copy()
        print("Selected program: Refresh")
        break
    else:
        print("You don't have choose a valid program. Please, give me a valid letter.")

print (" ")
durata = len(prog)

```

Figure 10.8. Python program lines 97 - 128

```python
#if the program would necessarily finish today
if in_giornata == "s":

#if the program selected lasts too long in respect with the end time given
    if float(durata) > float(fine) - float(ora):
        durata_ore = float(durata) * float(0.25)
        h_prog = int(durata_ore)
        m_prog = (float(durata_ore) - int(h_prog)) * 100
        if int(m_prog) <= 25:
            m_prog = "15"
        elif int(m_prog) <= 50:
            m_prog = "30"
        elif int(m_prog) <= 75:
            m_prog = "45"
        else:
            m_prog = "00"
            h_prog = int(h_prog) + 1

        print ("There is not enough time to let the chosen program finish on the end time selected")
        print ("Please, restart me and choose a new end time considering that the program selected lasts "
               + str(h_prog) + ":" + str(m_prog) + ".")
        os.system("pause")
        sys.exit()

#if the program selected must be launched immediately to finish on time
    elif float(durata) == float(fine) - float(ora):
        print ("There is no time to waste! Start the dryer right away to get it done on time.")
        os.system("pause")
        sys.exit()

```

Figure 10.9. Python program lines 129 - 158

```python

#calculation to understand when it consumes less and communication of the results
    else:
        partenza_massima = int(fine) - int(durata)

        consumo_minimo = float(9999999999)
        for inizio in range(int(ora), partenza_massima, 1):
            consumo = 0
            for intervallo in range(int(inizio), int(durata) + int(inizio), 1):
                consumo = consumo + float(rete_oggi[intervallo]) + float(prog[int(intervallo) - int(inizio)])
            if float(consumo) < float(consumo_minimo):
                consumo_minimo = float(consumo)
                indice = int(inizio)

        start = float(indice) * float(0.25)
        h_start = int(start)
        m_start = (float(start) - int(h_start)) * 100

        if int(m_start) < 25:
            m_start = "00"
        elif int(m_start) < 50:
            m_start = "15"
        elif int(m_start) < 75:
            m_start = "30"
        else:
            m_start = "45"

        print("I suggest you to start the dryer with the program selected at " + str(h_start) + ":" + str(m_start) + ".")
        print(" ")

        ritardo = float(start) - float(ora_attuale)
        h_ritardo = int(ritardo)
        m_ritardo = (float(ritardo) - int(h_ritardo)) * 100

```

Figure 10.10. Python program lines 159 - 190

```python

        if int(m_ritardo) < 25:
            m_ritardo = "00"
        elif int(m_ritardo) < 50:
            m_ritardo = "15"
        elif int(m_ritardo) < 75:
            m_ritardo = "30"
        else:
            m_ritardo = "45"

        print("To do this, however, it is necessary to schedule a delayed departure of " + str(h_ritardo) + ":" + str(m_ritardo) + ".")


#if the program could finish tomorrow
if in_giornata == "n":
    rete_oggi.extend(rete_domani)

#calculation to understand when it consumes less and communication of the results
    partenza_massima = int(fine) - int(durata) + ( int(int(len(rete_oggi)) / int(2)) )

    consumo_minimo = float(9999999999)
    for inizio in range(int(ora), partenza_massima, 1):
        consumo = 0
        for intervallo in range(int(inizio), int(durata) + int(inizio), 1):
            consumo = consumo + float(rete_oggi[intervallo]) + float(prog[int(intervallo) - int(inizio)])
        if float(consumo) < float(consumo_minimo):
            consumo_minimo = float(consumo)
            indice = int(inizio)

    start = float(indice) * float(0.25)

    print(" ")
```

Figure 10.11. Python program lines 191 - 222

```python
#if the starting time is on the next day
    if float(start) >= 24:
        start = float(start) - float(24)
        h_start = int(start)
        m_start = (float(start) - int(h_start)) * 100

        if int(m_start) < 25:
            m_start = "00"
        elif int(m_start) < 50:
            m_start = "15"
        elif int(m_start) < 75:
            m_start = "30"
        else:
            m_start = "45"

        ritardo = float(start) - float(ora_attuale) + float(24)
        h_ritardo = int(ritardo)
        m_ritardo = (float(ritardo) - int(h_ritardo)) * 100

        if int(m_ritardo) < 25:
            m_ritardo = "00"
        elif int(m_ritardo) < 50:
            m_ritardo = "15"
        elif int(m_ritardo) < 75:
            m_ritardo = "30"
        else:
            m_ritardo = "45"

        print("I suggest you to start the dryer with the program selected at " + str(h_start) + ":" + str(m_start) + " tomorrow.")
        print(" ")
        print("To do this, however, it is necessary to schedule a delayed departure of " + str(h_ritardo) + ":" + str(m_ritardo) + ".")
        print(" ")
        print("I wish you a good night and a peaceful rest.")

```

Figure 10.12. Python program lines 223 - 256

```python
#if the starting time is today
    else:
        h_start = int(start)
        m_start = (float(start) - int(h_start)) * 100

        if int(m_start) < 25:
            m_start = "00"
        elif int(m_start) < 50:
            m_start = "15"
        elif int(m_start) < 75:
            m_start = "30"
        else:
            m_start = "45"

        ritardo = float(start) - float(ora_attuale)
        h_ritardo = int(ritardo)
        m_ritardo = (float(ritardo) - int(h_ritardo)) * 100

        if int(m_ritardo) < 25:
            m_ritardo = "00"
        elif int(m_ritardo) < 50:
            m_ritardo = "15"
        elif int(m_ritardo) < 75:
            m_ritardo = "30"
        else:
            m_ritardo = "45"

        print("I suggest you to start the dryer with the program selected at " + str(h_start) + ":" + str(m_start) + ".")
        print(" ")
        print("To do this, however, it is necessary to schedule a delayed departure of " + str(h_ritardo) + ":" + str(m_ritardo) + ".")

#final greetings
print(" ")
print("Hope to have been helpful and to see you soon.")
print(" ")
print("Goodbye")
os.system("pause")
```

Figure 10.13. Python program lines 257 - 293

Conclusions

The work that I have done so far and that is described in this thesis, proposes the basis for the design and construction of a new electronic device that would meet the needs of power grid operators. The designed device would be able to control the turning on or off of a non-priority appliance in relation to the state of the electrical grid parameters. If this component were to be deployed on a large scale, it would therefore be possible to raise or lower even significantly the overall power consumption of the grid. As it has been described in this document, it can only be associated with one appliance and in the installation phase it must be adapted to its specific characteristics, although the control logic possesses two general common structures for the two categories of loads identified. In order to demonstrate this, practical tests have been carried out on appliances chosen as models, which have given positive results, thus demonstrating the goodness of the control logics I have implemented on the systems built. It is also marked on the fact that, being this a block logic that bases its choices on thresholds and time delays, in order not to cause too big instantaneous variations of power required to generators, they cannot be the same for all the devices produced. In the future it will therefore be necessary to study which distributions of times or thresholds should be assigned in order to obtain the most advantageous overall behavior for the purpose of enhancing the primary frequency control. Once this is done, it will also be necessary to streamline the hardware component by creating a new product that is cheaper than the device used in this design phase. This is because the Ekip UP Pro contains much more functionality than is needed to implement this control.

To conclude, although it was not the protagonist of this thesis work, an integrative tool was also proposed in the form of an application aimed at the end user, which can help to avoid the emergence of situations of difficult management of the electrical network, using the function of the delayed start of programmable home appliances. This product can be and must certainly be improved by implementing a machine learning component both on the prediction of domestic electricity consumption and on the prediction of generation from renewable sources, so as to be able to flatten as much as possible the power curve generation of the traditional power plants.

Bibliography

[1] European Commission. *Pathway to a Healthy Planet for All EU Action Plan: 'Towards Zero Pollution for Air, Water and Soil'*. Brussels, 12 May 2021.

[2] Gestione Energetica Ambientale S.r.l. *Renewable energies, Bloomberg forecasts for 2021.* 25 January 2021.

[3] Terna S.p.A. e Gruppo Terna. *Contesto ed evoluzione del sistema elettrico*. 2019.

[4] F. Baccino, F. Conte, F. Silvestro, S. Massucco, E. Ciapessoni, D. Cirio. *Impact Analysis of Load Control for Frequency Regulation: the case of Sardinia in 2020.* 2014 IEEE PES Innovative Smart Grid Technologies, Europe, 2014, pp. 1-6.

[5] F. Conte, M. Crosa di Vergagni, S. Massucco, F. Silvestro, E. Ciappessoni, D. Cirio. *Performance Analysis of Distributed Control Strategies for Frequency Regulation Support by Domestic Thermal Loads*. 2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI), 2018, pp. 1-6.

[6] A.Paolucci. *Lezioni di Trasmissione dell'energia elettrica.* 1998 Padova, CLEUP, 1998.

[7] Terna S.p.A. *Real Time Electricity Demand. www.terna.it.* 14 August 2021.

[8] . M. Eremia, M. Shahidehpour, L. Toma. *Handbook of electrical power system dynamics*. February 2013, pp. 137-178.

[9] Terna S.p.A. *Partecipazione alla regolazione di frequenza e frequenza-potenza.* 20 July 2008.

[10] E. Mallada, C. Zhao, S. Low. *Optimal load-side control for frequency regulation in smart grids.* 2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton), 2014, pp. 731-738.

[11] ENTSO-E. *ENTSO-E Network Code on Demand Connection.* 2012 Tech. Rep., Dec. 2012.

[12] Wikipedia - The Free Encyclopedia. *IP address.* $https://it.wikipedia.org/wiki/I\%C2\%B2Ccite_note - 7$

[13] Wikipedia - The Free Encyclopedia. *IP address.* *https://en.wikipedia.org/wiki/I%C2%B2C*

[14] J. Mankar, C. Darode, K. Trivedi, M. Kanoje, P. Shahare. *Review of I2C protocol.* January 2014.

[15] S. Campbell. *Basics of the I2C communication protocol.* 2016.

[16] Abb S.p.A. *Ekip UP The low voltage digital unit for next generation of plants.* 2020.

[17] AEG. *IP address.* *https://www.aeg.it/laundry/laundry/dryers/heat-pump-dryer/t8dec94st/*

[18] Hotpoint/Ariston. *IP address.* *https://www.hotpoint.it/h/forno-elettrico-incasso-hotpoint-colore-inox/859991548950*

[19] Abb S.p.A. *Ekip T e P test and programming unit.* 2020.

[20] ABB S.p.A. *SACE Emax 2 - Ekip Signalling 4K.* 2016.

[21] Energy Hunters *Fotovoltaico e batterie su abitazione – l'esempio tipico di abitazione in Italia* 7 October 2014.